Commercio elettronico: tecnologie per garantire la sicurezza

Salvatore Orlando

Dip. di Matematica Applicata ed Informatica
Università "Ca' Foscari" di Venezia
Via Torino, 155 - Venezia Mestre
email: orlando@unive.it

Sommario. Questo articolo si occupa delle problematiche di sicurezza nei sistemi di telecomunicazioni, con particolare riguardo alle problematiche delle transazioni elettroniche di tipo commerciale. Le tecnologie impiegate per garantire la sicurezza sono analizzate in dettaglio, in particolar modo quelle legate all'impiego di tecniche crittografiche. Infine si analizzano alcune iniziative correnti nell'ambito della sicurezza.

Indice

1. Introduzione....3
1. La sicurezza nei sistemi di telecomunicazione....4
1.1. Messaggi EDI e sicurezza....4
1.2. Attacchi ad un MHS per EDI....5
1.3. Requisiti di sicurezza....6
2. Soluzioni basate sulla crittografia....8
2.1. Crittologia, algoritmi, computer....8
2.2. Modello di critto-sistema....9
2.3. Esempio di attacco ad un critto-sistema....10
2.4. Schemi di crittografia a chiave simmetrica....10
2.5. Schemi di crittografia a chiave pubblica o asimmetrica....12
3. Esempio di transazione EDI sicura....21
3.1. Confidenzialità e autenticazione del messaggio....21
3.2. Schema di transazione sicura....23
3.3. Non ripudiabilità di invio e ricezione....24
3.4. Integrità del messaggio e della sequenza dei messaggi....24
4. Iniziative correnti per la sicurezza del commercio elettronico....25
4.1. ISO/OSI (X.400)....25
4.2. Internet (servizio di posta elettronica)....26
4.3. Il server commerciale sicuro di Netscape....26
4.4. Elettronic cash (Visa e Microsoft)....27
4.5. Sicurezza nello standard EDIFACT....27
5. Riferimenti....28

1. Introduzione

L'uso delle reti telematiche per le transazioni elettroniche, ovvero per lo scambio di dati attraverso le infrastrutture telematiche, pone dei problemi di *sicurezza* per i Sistemi Informativi Aziendali.

E' noto che un sistema informativo è sicuro nella misura in cui sono sicuri ed affidabili le persone che accedono ad esso. Molti sistemi aziendali non sono in rete, sono acceduti da un numero limitato di persone che non hanno interesse a manometterne il funzionamento, e sono oggetto di regolari *backup* per garantire la protezione del sistema da guasti hardware/software.

Una prima esigenza di sicurezza sorge quando un'azienda vuole rendere confidenziali alcuni dati per permetterne la consultazione ad un numero limitato di suoi dependenti. A questo punto si rende necessario introdurre un sistema di chiavi per regolamentare l'accesso al sistema (*accounting*) o per cifrare i dati confidenziali (*crittografia*). Un primo problema è quindi generare un sistema di *chiavi sicure*, ovvero chiavi difficilmente individuabili senza l'aiuto di persone che conoscono tali chiavi e che, volontariamente o involontariamente, le rendono pubbliche. L'altro problema è rendere sicura la *comunicazione delle chiavi*. Ad esempio, se due utenti devono leggere dei dati cifrati con una certa chiave, hanno necessità di scambiare e mantenere aggiornata la conoscenza comune sulla chiave impiegata. Tale scambio di informazioni deve essere protetto, per evitare che qualche malintenzionato acceda alla chiave e quindi alle informazioni confidenziali (*badge* e *smart card* sono metodi per distribuire in maniera più sicura delle chiavi di accesso).

La connessione in rete di un sistema informativo aziendale, necessaria per permettere lo scambio di messaggistica strutturata (EDI) e non, pone ulteriori problemi di sicurezza. La più importante caratteristica di sicurezza che un sistema del genere deve garantire riguarda la *confidenzialità* della messaggistica in ingresso/uscita ai sistemi informativi aziendali. L'altro aspetto di sicurezza riguarda l'*integrità* dei messaggi trasmessi, che tra l'altro comprende la protezione contro la perdita o la duplicazione dei messaggi. Altri aspetti di sicurezza, non meno importanti, sono l'*autenticazione del mittente* dei messaggi, e la *non repudiabilità di invio/ricezione dei messaggi* da parte del mittente/destinatario. Molte di queste caratteristiche di sicurezza sono garantite attraverso l'uso di una tecnologia innovativa per la cifratura delle informazioni, nota come *sistema di crittografia a chiave pubblica*. Questa tecnologia permette di *firmare elettronicamente i messaggi*, e permette quindi ad un'*autorità* riconosciuta di garantire, attraverso la sua firma elettronica, i *certificati elettronici* (o *credenziali*) dei partner di una comunicazione. I certificati contengono i dati di un individuo/azienda e la sua cosiddetta chiave pubblica, utilizzata dal sistema di crittografia.

1. La sicurezza nei sistemi di telecomunicazione

Per rendere sicuro un qualsiasi sistema è necessario analizzare e classificare i possibili attacchi alla sua sicurezza. Si noti che l'approccio è simile a quello seguito per migliorare la tolleranza ai guasti di un sistema, dove è necessario analizzare attentamente i possibili guasti al sistema per poter approntare un insieme di soluzioni adeguate.

Un sistema di gestione di messaggistica strutturata (EDI) e non, soprattutto se aperto come Internet, si presta ad un insieme molto vasto di attacchi, che coinvolgono le linee di comunicazione o i nodi intermedi attraversati dalle comunicazioni.

In questa sezione sono analizzati alcuni di questi attacchi, e sono individuati i requisiti di sicurezza minimi per garantire la protezione della messaggistica in transito da questo tipo di attacchi. Anche se l'analisi è rivolta a sistemi per lo scambio di messaggi in formato EDI, la stessa analisi si applica ad altri sistemi per lo scambio di informazioni attraverso le retei di telecomunicazione.

1.1. Messaggi EDI e sicurezza

Un sistema per la trasmissione di transazioni EDI poggia su un sistema, solitamente più generale, per la trasmissione di messaggi chiamato Message Handling System (MHS). Esistono MHS aperti, come X400 o SMTP/Internet, o chiusi, come i VAN (Value Added Networks).

Un VAN fonda molto del proprio sistema di sicurezza sulla protezione, anche fisica, delle proprie macchine e linee di comunicazione interne. Questo garantisce agli utenti che la trasmissione dei messaggi avvenga in maniera sicura. La protezione del VAN nei confronti dell'esterno si basa sull'autenticazione degli utenti che vogliono accedere al VAN per inviare o ricevere posta elettronica. L'autenticazione, che avviene tramite scambio di password all'atto del collegamento del singolo utente al VAN, garantisce anche sull'origine dei messaggi stessi. Si noti come, in questo caso, il certificatore dell'autenticità dei messaggi è il VAN stesso.

Un ulteriore livello di sicurezza, necessario per proteggere la messaggistica dell'utente, è la trasmissione codificata del traffico in transito sul collegamento (di solito dial-up) tra il computer dell'utente e il VAN.

Si noti come queste forme di sicurezza non proteggono l'utente dal VAN stesso, a meno che il traffico tra coppie di utenti venga reso segreto tramite opportuna codifica crittografica (questo richiede un preventivo accordo tra utenti, indipendentemente dal VAN). In ogni caso, è necessario adottare tecnologie di sicurezza che permettano ai VAN di interagine (Internetworking), e quindi di *aprirsi* versi l'esterno. Le tecnologie da adottare sono quelle già studiate per i cosiddetti *sistemi aperti*.

I problemi di sicurezza nei sistemi aperti sono molto gravi, in quanto non è possibile fare affidamento sulla sicurezza del sistema di comunicazione: le linee e i nodi intermedi sono da considerarsi non sicuri.

Nel seguito faremo soprattutto riferimento alle problematiche di *sicurezza nei sistemi aperti*

1.2. Attacchi ad un MHS per EDI

In questa sezione descriveremo i principali tipi di attacco ad un MHS aperto che trasporta messaggistica EDI. Figura 1 illustra un MHS composto da

- UA (User Agent, ovvero il sistema che interfaccia l'utente)
- MS (Message Store, il sistema che mantiene la mailbox dell'utente con i messaggi in ingresso e in uscita)
- MTS (Message Transfer System, il sistema di rete che si occupa dell'inoltro dei messaggi secondo uno schema store-and-forward). L'MTS è costituito da un insieme di MTA (Message Transfer Agents) attraverso cui avviene l'instradamento dei messaggi.

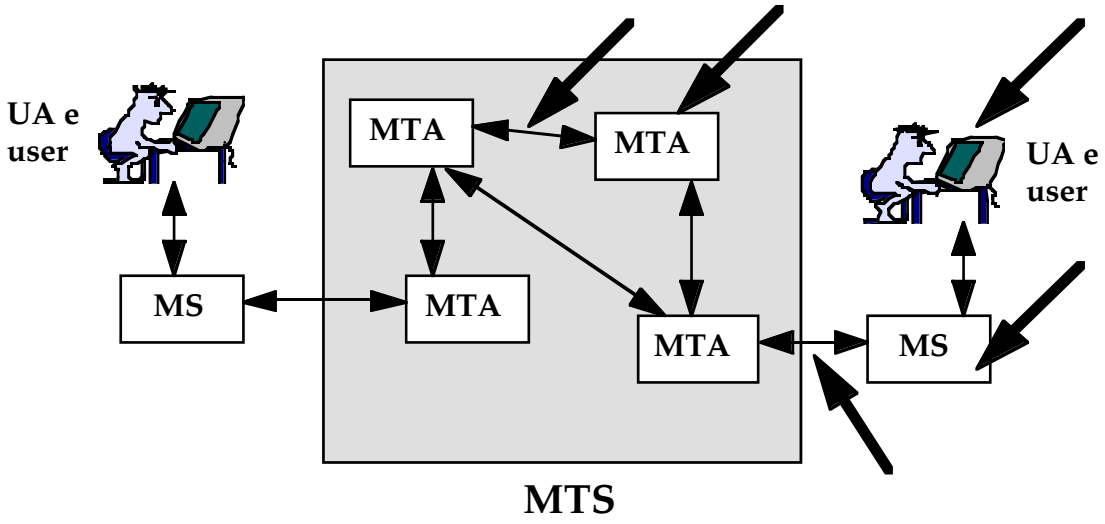

Figura 1. Un MHS che trasporta messaggistica EDI.

Un MHS, che supponiamo essere in grado di trasportare messaggistica EDI può subire diversi attacchi. Se il sistema è aperto, tutti i possibili attacchi devono essere presi in considerazione per approntare opportune difese. Un'attacco può essere portato alle connessioni o ai sistemi attraversati dalle comunicazioni. In Figura 1 sono stati esemplificati parecchi punti del sistema nei cui confronti un attacco potrebbe essere portato. Probabilmente

l'attacco più pericoloso è quello portato al computer dell'utente (UA), che richiede una vera e propria protezione fisica.

Di solito si presuppone che gli attacchi portati all'MHS siano i più svariati e tutti possibili. Si presuppone, quindi, che malintenzionati possano "ascoltare" il traffico in transito, inviare nuovi messaggi sulla rete, bloccare la trasmissione di altri messaggi, ecc. Alcuni dei possibili attacchi sono i seguenti:

- lettura dei messaggi;
- invio illegale di messaggi;
- alterazione dei messaggi in transito;
- cancellazione, duplicazione, o alterazione della sequenza dei messaggi;
- blocco o alterazioni di una linea di comunicazione.

I requisiti di sicurezza di un sistema EDI devono garantirne la protezione della trasmissione anche in presenza di tutti questi tipi di attacchi.

1.3. Requisiti di sicurezza

Nella sezione precedente sono stati elencati i tipi di attacco che possono essere portati ad un MHS. Anche in presenza di questi attacchi, un MHS che trasporta messaggistica EDI (nel seguito indicato come MHS-EDI) deve garantire i seguenti *requisiti di sicurezza*:

- *Confidenzialità* del contenuto dei messaggi, ovvero il vero contenuto dei messaggi devono essere protetti da letture non autorizzate;
- *Autenticazione del mittente* dei messaggi, ovvero l'origine dei messaggi deve essere controllabile inconfutabilmente per evitare l'invio illegale di messaggi da parte di malintenzionati
- *Integrità del contenuto* dei messaggi, ovvero il destinatario deve poter controllare che il messaggio ricevuto non è stato manomesso durante la trasmissione;
- *Integrità della sequenza* dei messaggi, ovvero non deve essere possibile alterare l'ordine dei messaggi, o inviare o duplicare dei messaggi senza che il destinatario si renda conto del malfunzionamento;
- Impossibilità di determinare il contenuto delle trasmissioni dalla frequenza e dal tipo di traffico.

Vedremo come, attraverso l'uso della *crittografia*, sia possibile ottenere molti dei requisiti sopra elencati.

L'ultimo, il requisito di sicurezza che cerca di impedire deduzioni statistiche sul tipo di traffico, richiede che l'MHS instradi i messaggi su più linee di comunicazione (un malintenzionato potrebbe essere in ascolto su una sola di queste linee), e che alcuni messaggi finti, opportunamente codificati, vengano inviati tra messaggi veri (traffic padding).

Oltre ai requisiti di sicurezza sopra elencati, ne esistono altri che garantiscono non solo da attacchi esterni, ma anche da usi scorretti dell'MHS-EDI da parte degli stessi partner di una comunicazione. Questi requisiti sono:

- *Non ripudiabilità dell'invio* di un messaggio, ovvero non deve essere possibile, per il mittente di un messaggio, ripudiare l'invio del messaggio stesso;
- *Non ripudiabilità della ricezione* di un messaggio, ovvero non deve essere possibile, per il destinatario di un messaggio, ripudiare la ricezione del messaggio stesso.

Per ottenere questi ultimi requisiti di sicurezza deve essere possibile coinvolgere anche terze parti per verificare che un messaggio giunto ad un certo soggetto destinatario sia stato effettivamente originato da un certo mittente (*non ripudiabilità*). Questo vale anche i messaggi di ricevuta di ritorno, necessari per realizzare la non repudiabilità della ricezione.

2. Soluzioni basate sulla crittografia

Il metodo principale per garantire la sicurezza di un MHS-EDI e, in generale, la sicurezza e la privatezza delle comunicazioni, è la *crittografia*. La crittografia si occupa di rendere *cifrato*, e quindi segreto, un testo che altrimenti sarebbe in chiaro e quindi leggibile da tutti. La *crittografia* è basata sull'esistenza di chiavi per la cifratura. Esistono due schemi principali per crittografare. Il primo, che è anche il può antico, è detto a *chiave privata*. Lo schema più recente è detto *a chiave pubblica*. Mentre il primo schema è di solito usato solo per garantire la confidenzialità dei messaggi, il secondo è largamente impiegato per garantire tutti gli altri requisiti di sicurezza elencati nella sezione precedente. L'applicazione più importante dello schema a chiave pubblica è la *firma elettronica* dei messaggi. La disciplina complementare alla crittografia è detta *crittoanalisi*. Questa disciplina si occupa di decifrare i testi crittografati, e alcuni dei metodi impiegati da persone che vogliono portare attacchi alla sicurezza di un sistema sono appunto basati sui risultati della crittoanalisi. Si noti che, nel momento in cui si impiega un certo schema di crittografia, si presuppone che gli avversari possano impiegare i risultati più avanzati di crittoanalisi, ovvero il massimo attacco possibile portato allo schema di crittografia.

Si noti che la *crittoanalisi* non è una disciplina a sè stante, impiegata solo da malintenzionati. Qualunque schema di crittografia deve essere infatti validato, dimostrando che esso è *difficilmente* compromettibile da parte di un crittoanalista. La crittoanalisi è quindi fondamentale per validare la sicurezza di un critto-sistema.

Le origini della crittologia sono antichissime: esistono esempi d'uso della crittografia che risalgono agli antichi Romani. L'uso della crittologia nella seconda guerra mondiale è stata fondamentale, e grande è stata la perizia dei crittoanalisti alleati nel decifrare le comunicazioni crittografate dell'esercito nazista.

Oggi la crittologia viene impiegata largamente per scopi civili, per garantire la privatezza delle comunicazioni e dei dati memorizzati su supporti elettronici/magnetici.

2.1. Crittologia, algoritmi, computer

Oggi, con l'avvento dei computer, è diventata chiara la relazione tra *crittologia* e *informatica*, soprattutto con la branca dell'informatica che si occupa di *algoritmi* e *complessità*

Un metodo di crittografia/crittoanalisi è essenzialmente un *algoritmo* che, in quanto tale, può essere eseguito realizzato tramite un programma ed eseguito da un computer. La *complessità di un algoritmo* è una misura del tempo impiegato per la sua risoluzione da parte di un computer.

I computer sono strumenti potenti per *crittografare* e *de-crittografare*, anche se l'uso di sistemi *molto* potenti è soprattutto necessaria per la *crittoanalisi*.

I sistemi di crittografia sicuri di solito richiedono *algoritmi di crittoanalisi* di *complessità esponenziale* (algoritmi che necessitano di anni di calcolo anche su sistemi paralleli). Un sistema crittografico *sicuro* è appunto quello che necessita di algoritmi di crittoanalisi tanto complessi da richiedere tempi dell'ordine di anni anche impiegando computer molto potenti.

2.2. Modello di critto-sistema

Un modello di crittosistema, ovvero un modello che usa la crittografia per proteggere le comunicazioni, è basato solitamente su due algoritmi, rispettivamente per la crittografia e la decrittografia. Questi algoritmi hanno a comune la conoscenza delle chiavi necessarie per rendere cifrato il testo del messaggio e per riportare il testo in chiaro.

Come si può desumere da modello illustrato in Figura 2, poiché le linee di comunicazione sono considerate *non sicure*, si presuppone che il crittoanalista (malintenzionato) possa accedere al testo cifrato, ma non al sistema di chiavi. Inoltre si presuppone che questi sia a conoscenza degli algoritmi usati per crittografare e decrittografare. La *segretezza del sistema delle chiavi* è quindi l'unica garanzia sulla sicurezza del critto-sistema.

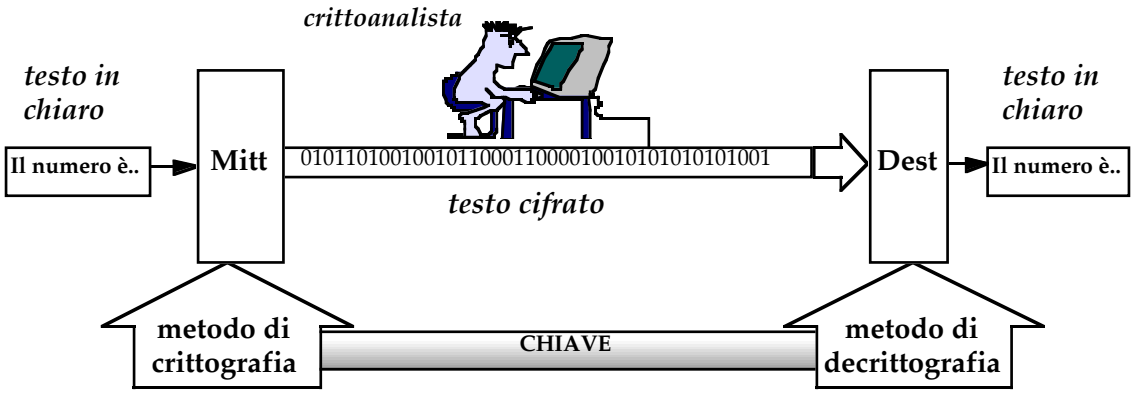

Figura 2. Schema di critto-sistema.

Il fattore *costo computazionale* è importante nell'assicurare i requisiti di sicurezza di un critto-sistema. Mentre gli algoritmi di critto e decrittografia devono essere poco costosi (ovvero, devono poter essere eseguiti in pochissimo tempo da un computer) poiché mittente e destinatario devono poterli eseguire velocemente ad ogni trasmisione, il costo degli algoritmi per decrittografare i messaggi senza la conoscenza preventiva delle chiavi deve essere non sostenibile da parte del crittoanalista.

2.3. Esempio di attacco ad un critto-sistema

Un esempio di attacco che un *crittoanalista* può portare ad un critto-sistema può essere basato sulla conoscenza, per un certo testo cifrato acceduto in maniera non autorizzata, del corrispondente testo in chiaro. Grazie alla conoscenza dell'algoritmo di crittografia, il crittoanalista può quindi cercare di individuare la chiave per prove ed errori.

Se il testo in chiaro non è noto, esso può essere indovinato, oppure può essere dedotto dal fatto che certe parti del messaggio sono standard (intestazione del messaggio, indirizzo del mittente, ecc.).

L'attacco al crittosistema descritto precedentemente è solo uno dei possibili attacchi. Lo scopo principale dell'attacco è sempre la determinazione della chiave di crittografia. Conoscendo la chiave, sarebbe poi possibile decifrare tutto il resto del traffico senza impiegare i costosi algoritmi di crittoanalisi.

Un modo per rendere poco utile lo sforzo di individuare la chiave è quello di modificare in continuazione la chiave di codifica. E' possibile, come vedremo, generarne una per sessione, e in maniera casuale.

2.4. Schemi di crittografia a chiave simmetrica

In questa sezione ci occuperemo dei cosiddetti schemi di crittografia a *chiave simmetrica*. Tale denominazione deriva dalla proprietà che *una singola chiave* è usata sia per la crittografia (dal testo in chiaro a quello cifrato) che per la decrittografia (dal testo cifrato a quello in chiaro). Lo schema prevede che le funzioni di crittografia e decrittografia siano pubbliche, per cui la sicurezza è soltanto garantita dalla segretezza della chiave simmetrica.

Le funzioni di crittografia e decrittografia sono di solito *poco costose dal punto di vista computazionale*. Questo schema è quindi usato per rendere confidenziali grosse quantità di dati.

Il problema più grosso da affrontare è però la distribuzione delle chiavi, soprattutto a causa del numero elevato di chiavi necessarie per ottenere comunicazioni sicure tra tutte le possibili coppie di partner di comunicazione. Un altro problema legato alla sicurezza riguarda l'impossibilità di provare a terze parti l'autenticità di un messaggio crittografato con una chiave simmetrica.

L'algoritmo di crittografia simmetrico più famoso è il DES, che sta per Data Encryption Standard, ed è certificato dall'amministrazione degli U.S.A.

Crittosistemi simmetrici

Un crittosistema simmetrico è semplicemente basato su

- una funzione di *crittografia*: *C()*

- una funzione di *decrittografia*: *D()*
- una singola *chiave Key*, impiegata sia da *C()* e sia da *D()*.

Se chiamiamo *P* il *plaintext* (ovvero il *testo in chiaro*), allora il *ciphertext*, ovvero il *testo cifrato*, è dato da:

C(Key, P)

mentre il *plaintext* si può riottenere come:

D(Key, C(Key, P))

In Figura 3 viene illustrata la trasformazione da testo in chiaro a testo cifrato e viceversa effettuata, rispettivamente, dalla funzione *C()* e *D()* impiegando la medesima chiave simmetrica.

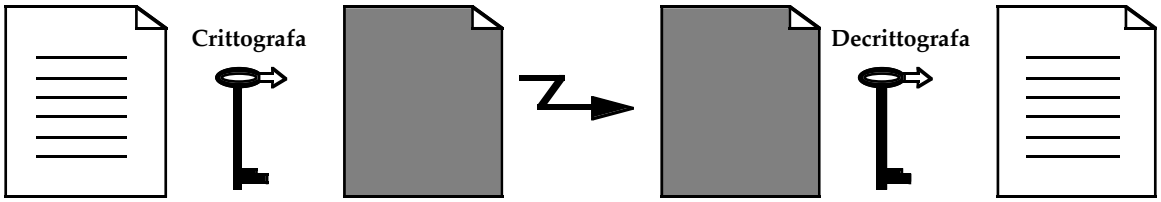

Figura 3. Crittografia e decrittografia a *chiave simmetrica*.

I crittosistemi simmetrici hanno diversi problemi:

- è necessario un canale sicuro per lo scambio delle chiavi tra ogni coppia di partner che vogliono scambiarsi messaggi;
- lo scambio delle chiavi deve precedere ogni comunicazione;
- per garantire la segretezza di ogni singola comunicazione, è necessaria una chiave diversa per ogni canale di comunicazione. Sono quindi necessarie n^2 chiavi differenti per avere una chiave diversa per tutte le possibili comunicazioni tra n differenti partner;
- non è possibile dimostrare a terzi, non coinvolti direttamente nella comunicazione tra due partner, l'identità del mittente di un messaggio. Per ottenere questo bisognerebbe, infatti, rivelare la chiave simmetrica usata per la comunicazione. Inoltre, proprio perché la chiave usata è simmetrica, il messaggio crittografato potrebbe essere stato generato proprio dal destinatario.

Alcuni di questi problemi rendono i crittosistemi simmetrici non adatti a garantire alcuni dei requisiti di sicurezza di un sistema MHS-EDI. Ad esempio, non è possibile garantire la non ripudibilità dell'invio del messaggio, in quanto abbiamo visto che non è possibile coinvolgere terze parti nella verifica dell'origine del messaggio.

DES, un algoritmo di crittografia simmetrico

DES, che sta per Data Encryption Standard, è un algoritmo di crittografia simmetrico certificato dall'amministrazione degli U.S.A. DES è usato per la protezione dei dati nelle transazioni finanziarie, ma ne esiste anche una versione più sicura per applicazioni militari.

DES è stato sviluppato da ricercatori IBM, ed è identico al Data Encryption Algorithm (DEA) accettato come standard ANSI. Esso crittografa blocchi di 64 bit usando una chiave di 56 bit, usando semplici operazioni iterate per un certo numero di volte. L'algoritmo DES base può però essere applicato a blocchi più grossi di 64 bit usando schemi standard diversi, con differenti livelli di sicurezza. Ad esempio, lo schema *ECB* (Electronic Code Book) prevede che DES sia applicato separatamente ai vari blocchi di 64 bit che compongono un testo.

L'attacco al DES, soprattutto se portato ad un testo cifrato usando lo schema ECB è abbastanza semplice. Il crittoanalista può collezionare ricorrenze di codici, e confrontare queste ricorrenze con quelle comunemente incontrate nei testi in chiaro in una certa lingua. La chiave del DES, inoltre, è molto corta (solo 2^{56} chiavi possibili), rendendo possibili tecniche basate sulla *forza bruta,* ovvero su prove ed errori impiegando grosse potenze di calcolo. Da notare, a questo riguardo, la polemica che ha accompagnato la nascita dello standard (anni 77-78), con l'intervento del National Security Agency per ridurre a 56 bit la dimensione della chiave. La NSA è stata sospettata di voler permettere al governo, che dispone di macchine sufficientemente potenti, di decrittografare grazie alla *forza bruta* un qualsiasi testo cifrato.

Esistono schemi standard di applicazione di DES migliori di ECB, che eliminano il fenomeno della ricorrenza e/o impiegano più chiavi. Il livello di sicurezza del DES aumenta se gli schemi di applicazione impiegati sono accompagnati da cambiamenti continui della chiave simmetrica.

2.5. Schemi di crittografia a chiave pubblica o asimmetrica

Gli schemi di crittografia a chiave pubblica o asimmetrica risolvono molti dei problemi tipici dei sistemi a chiave simmetrica. Si ricordi che i maggiori problemi di questi ultimi riguardano la comunicazione delle chiavi, e l'autenticazione dei messaggi anche nei confronti di soggetti terzi. Gli schemi di crittografia asimmetrica risolvono egregiamente questi problemi. Il loro nome deriva dall'impiego di *coppie di chiavi distinte,* una usata per crittografare e l'altra per decrittografare. Ogni soggetto sceglie un'opportuna coppia di chiavi, mantiene *segreta* una di queste chiavi, e rende *pubblica* l'altra. Si noti che non è possibile derivare la chiave privata da quella pubblica, per cui quest'ultima può essere comunicata a tutti partner interessati. Proprio questa proprietà semplifica il problema delle comunicazioni delle chiavi. Inoltre, non abbiamo il problema della proliferazione delle chiavi che avevamo osservato per le chiavi simmetriche.

Si noti che, anche in questo caso, si assume che le funzioni di crittografia e decrittografia siano pubbliche. La sicurezza è quindi garantita dalla sola segretezza della chiave privata.

Gli schemi a chiave pubblica sono usati per rendere confidenziali i dati di una comunicazione, ma forse la loro più importante applicazione è la *firma elettronica* dei messaggi. In questo caso, però, le due chiavi, pubblica e privata, sono usate in maniera inversa

rispetto al loro uso per garantire la confidenzialità. Ciò è possibile grazie alla seguente proprietà degli schemi a chiave pubblica: tutto ciò che è crittografato usando una qualsiasi delle due chiavi, può essere decrittografato usando la rimanente chiave.

Le funzioni di crittografia e decrittografia sono di solito *molto costosi dal punto di vista computazionale*, ma il livello di sicurezza nei confronti degli attacchi dei crittoanalisti è molto elevata. A causa del loro elevato costo computazionale, quando questi schemi di crittografia sono usati per scopi di confidenzialità, essi sono applicati a *dati di piccole dimensioni.*

Il più conosciuto crittosistema a chiave pubblica è RSA, dal nome dei suoi inventori Rivest, Shamir e Adleman. Il metodo, proposto nel 1978, è attualmente coperto da un brevetto. Inoltre, il governo americano ritiene non esportabile tale tecnologia (i codici per la crittografia sono considerati assimilabili ad armamenti, e quindi non esportabili senza il consenso delle autorità).

Crittosistemi asimmetrici

Un crittosistema asimmetrico che vuole garantire la confidenzialità della trasmissione di un certo testo in chiaro è basato su:

- una funzione di *crittografia*: *C()*
- una funzione di *decrittografia*: *D()*
- una coppia di *chiavi* K_{pub} e K_{priv}

Se chiamiamo *P* il *plaintext* (ovvero il *testo in chiaro*), il *ciphertext*, ovvero il *testo cifrato*, è dato da:

$$C(K_{pub}, P)$$

mentre il *plaintext* si può riottenere come:

$$D(K_{priv}, C(K_{pub}, P))$$

In Figura 4 viene illustrata la trasformazione da testo in chiaro a testo cifrato e viceversa effettuata, rispettivamente, dalla funzione *C()* e *D()* impiegando le due chiavi.

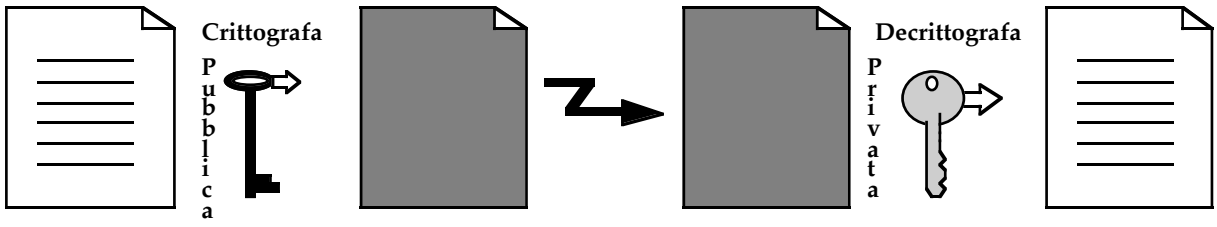

Figura 4. Crittografia e decrittografia a *chiavi asimmetriche*.

La coppia di chiavi impiegata è quella associata col destinatario del messaggio. Il mittente usa la chiave pubblica del destinatario per crittografare, e il destinatario è l'unico soggetto in grado di decrittografare il messaggio tramite la propria chiave privata.

E' importante capire che tra le due chiavi *deve esistere una relazione matematica ben precisa*. La coppia di chiavi deve essere generata in modo che questa relazione sia rispettata, e la generazione potrebbe avvenire a cura del soggetto che poi deterrà come *segreta* una delle due chiavi.

Uno dei vantaggi dei crittosistemi asimmetrici rispetto a quelli simmetrici è la riduzione del numero di chiavi necessarie. Infatti, se *n* sono i soggetti che desiderano comunicare tra di loro, sono necessarie solo *n* coppie di chiavi, una per ogni soggetto. Per ogni coppia di chiavi, solo una, quella *pubblica*, viene resa disponibile a tutti.

Potremmo pensare alle chiavi pubbliche come a numeri telefonici, per cui diventa necessario un elenco pubblico e affidabile delle chiavi pubbliche. Vedremo nel seguito come è possibile distribuire in maniera sicura le chiavi pubbliche di ciascun utente. La sicurezza nella diffusione delle chiavi pubbliche non è secondaria. Infatti, un utente A potrebbe far finta di essere B, e comunicare una chiave pubblica *ad hoc* in modo da essere in grado di decrittografare tutti i messaggi inviati a B.

I crittosistemi asimmetrici sono molto sicuri, grazie soprattutto all'adozione di chiavi molto lunghe. Il loro più grande svantaggio è costituito dai costi computazionali elevati degli algoritmi di crittografia e decrittografia. Proprio quest'ultima proprietà sconsiglia l'uso di tali sistemi per crittografare/decrittografare dati di grosse dimensioni.

RSA, un crittosistema a chiave pubblica

RSA è un crittosistema a chiave pubblica proposto nel 1978, che deve il suo nome ai suoi inventori R.Rivest, A.Shamir e L.Adleman. Tale metodo è coperto da brevetto e non è esportabile al di fuori degli U.S.A., se non in una forma meno sicura (con chiavi più corte).

La sicurezza dello schema RSA dipende dalla segretezza della chiave *privata*, e dalla lunghezza delle chiavi (chiavi dell'ordine di *centinaia di bit*).

La lunghezza delle chiavi e l'algoritmo di crittografia usato rendono molto difficoltosa la derivazione della chiave privata, anche se si conosce sia la corrispondente chiave pubblica e sia il testo con essa crittografato. Infatti per risalire alla chiave pubblica sarebbe necessario *fattorizzare con una coppia di numeri primi* un numero molto grande. Questo compito è molto complesso, e prenderebbe anni anche sui più veloci calcolatori.

Esiste anche una società, RSA Data Security Inc, che rilascia il software. Tale società sponsorizza, con premi in denaro, sfide per individuare la fattorizzazione di numeri enormi. Tutto ciò per dimostrare che il metodo è sicuro, anche se i migliori crittoanalisti si impegnano spinti dal premio in danaro.

Firme elettroniche

La sicurezza sull'origine dei messaggi è spesso più importante della confidenzialità del messaggio stesso. Per garantire l'autentiticità di un messaggio è possibile "firmarlo elettronicamente".

La *firma elettronica* di un messaggio è ottenuta tramite un crittosistema a chiave pubblica, basato però sull'*uso inverso* delle coppie di chiavi (privata, pubblica) associate ad ogni utente rispetto all'uso illustrato precedentemente per garantire la confidenzialità. L'uso delle chiavi è infatti il seguente:

- il mittente che vuole firmare un proprio messaggio impiega la propria chiave *privata* per crittografare il messaggio stesso. Nota che questo non garantisce la confidenzialità del messaggio, in quanto *tutti* possono decrittografarlo usando la chiave pubblica del mittente.
  Il messaggio così crittografato diventa la *firma elettronica* che viene attaccata al messaggio stesso in chiaro. Messaggio e firma elettronica sono così inviati al destinatario;
- il destinatario riceve il messaggio e la firma elettronica, e decrittografa quest'ultima con la chiave *pubblica* del mittente. Confronta quindi il messaggio in chiaro con quello ottenuto decrittografando la firma elettronica. Il successo del confronto garantisce l'autenticità dell'origine del messaggio ricevuto: infatti, solo il soggetto in possesso della chiave privata corrispondente a quella pubblica avrebbe potuto produrre la firma elettronica ricevuta.

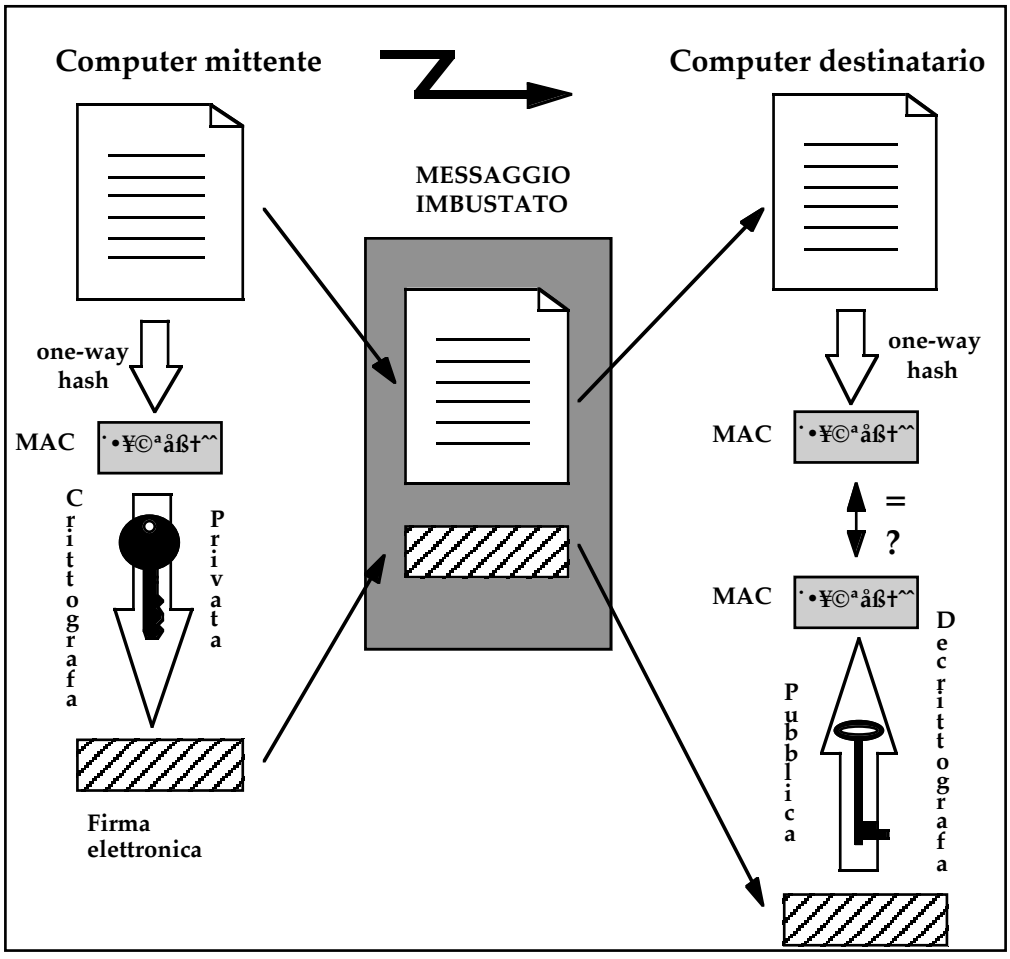

Figura 5. Trasmissione di un messaggio firmato elettronicamente.

Per firmare elettronicamente è possibile impiegare la stessa coppia di chiavi usata per garantire la confidenzialità della trasmissione, oppure una coppia differente. Nel seguito faremo riferimento alla seconda alternativa, ovvero supporremo che ad ogni utente siano associate due coppie di chiavi: la prima impiegata per scopi di confidenzialità, e la seconda impiegata per firmare elettronicamente. Ogni utente dovrà quindi pubblicare due diverse chiavi (pubbliche).

Lo schema illustrato precedentemente, ovvero la *firma elettronica* prodotta crittografando tutto il messaggio, non è però praticabile a causa dell'alta complessità computazionale degli algoritmi di crittografia/decrittografia a chiave pubblica. La firma elettronica viene in pratica ottenuta applicando lo stesso schema visto sopra ad un *messaggio condensato,* detto MAC (*Message Autentication Code*). Il MAC, ovvero un testo di lunghezza fissa ottenuto a partire da un testo a lunghezza variabile, è generato con algoritmi noti con il nome di funzioni *one-way hash*. E' quindi necessario impiegare algoritmi *affidabili* per generare il MAC, in modo che piccolissimi cambiamenti nel messaggio diano luogo a MAC completamente differenti. MD2, MD4 e MD5 sono sigle di tipici algoritmi che implementano funzioni *one-way hash*.

In Figura 5 viene illustrato uno schema di trasmissione che fa uso di un crittosistema a chiave pubblica per garantire *soltanto* l'autenticità del mittente (*firma elettronica*). Si noti come la funzione *one-way hash* deve essere nota a mittente e destinatario.

Anche l'uso illustrato in Figura 5 della crittografia a chiave pubblica non garantisce la confidenzialità del messaggio, che viaggia in chiaro. Si noti però che lo schema permette, oltre al controllo dell'autenticità dell'origine del messaggio, il controllo dell'integrità del messaggio stesso. Infatti, se il messaggio giungesse *non integro*, allora il controllo della firma elettronica fallirebbe.

Certificati elettronici

La firma elettronica garantisce l'autenticazione dei partner di una comunicazione, a patto che non ci siano dubbi sull'autenticità della *chiave pubblica* impiegata. E' quindi necessario trasmettere in maniera sicura e non manomettibile la chiave pubblica (o le chiavi pubbliche) associata ad un certo soggetto. Si potrebbe pensare che quest'ultimo invii *direttamente,* su un *canale sicuro,* la propria *chiave pubblica* a tutti i sui possibili destinatari. Questo metodo è in alcuni casi poco pratico, perché

- può diventare difficile individuare un *canale sicuro*;
- diventa complicato il primo contatto tra due partner di una comunicazione.

I *certificati elettronici* sono metodi semplici e sicuri per garantire la diffusione delle chiavi pubbliche. Il certificato lega i dati identificativi di un utente con la sua, o le sue, chiavi pubbliche. Il certificato è esso stesso firmato elettronicamente da un'*autorità* riconosciuta come affidabile da tutti i partner di una comunicazione.

Chiaramente le chiavi private non devono essere incluse nel certificato. Per garantire la sicurezza, neanche l'autorità che appone la firma sul certificato deve conoscere le chiavi private

corrispondenti a quelle pubbliche. Su questo punto è comunque aperto un dibattito. I governi, per ragioni di sicurezza nazionale, sono interessati a poter decifrare i messaggi crittografati. Senza questa capacità, ad esempio, l'autorità giudiziaria non potrebbe usare il controllo delle comunicazioni come mezzo investigativo.

In Figura 6 si illustra il processo di rilascio del certificato del soggetto Alice. Alice presenta un documento ufficiale con i suoi dati anagrafici e le chiavi da rendere pubbliche, l'autorità ha il solo compito di certificare la relazione tra dati identificativi e chiavi apponendo la sua firma elettronica. Si noti come l'autorità *non deve* conoscere le chiavi private di Alice.

A questo punto possiamo entrare più nel dettaglio, facendo un esempio di formato di un certificato elettronico. Quindi introduremo la problematica della distribuzione dei certificati, che può essere elegantemente risolta tramite una gerarchia di autorità per la certificazione. Infine tratteremo delle tecniche per conservare in tutta sicurezza la chiave privata associate a quella pubblica .

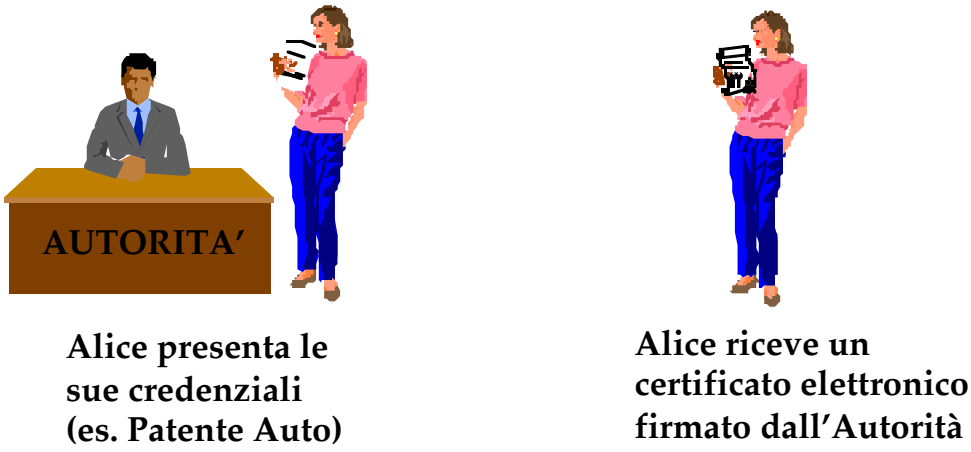

Figura 6. Rilascio del certificato elettronico di Alice.

Formato di un certificato elettronico

Il certificato è una struttura dati che serve per legare in maniera sicura una o più chiavi pubbliche con degli attributi identificativi. Il formato illustrato in Figura 7 è molto simile a quello definito dalla raccomandazione CCITT (*Comité Consultatif Int.ernational Télégraphique et Téléphonique*) X.509 e adottato da ISO (*International Standard Organization*).

Il campo VERSIONE identifica il formato del certificato (nel tempo le versioni potrebbero cambiare). Il NUMERO DI SERIE identifica il certificato tra quelli inviati dalla stessa autorità. L'ALGORITMO DI FIRMA indica l'algoritmo usato dal certificatore per generare il MAC (es.: MD2) e quello per generare la firma elettronica (es.: RSA).

I campi DATI DEL CERTIFICATORE e DATI DI ALICE riportano, in maniera *standard*, i dati identificativi e gli indirizzi di rete dell'autorità certificatrice e della persona a

cui il certificato si riferisce. E' importante che ci sia un modo *standard* ed *univoco* per identificare i soggetti coinvolti, anche perchè la firma digitale dipende dal formato di queste informazioni (X.500 è la raccomandazione CCITT per il servizio di naming degli oggetti che costituiscono un sistema di comunicazione. Ogni oggetto è caratterizzato da un DN, Distinguished Name).

Il campo VALIDITA' indica i limiti temporali di validità del certificato. Infine, abbiamo il campo che contiene le chiavi pubbliche di ALICE. La firma elettronica del certificatore è ottenuta a partire dal contenuto di tutti i campi precedenti.

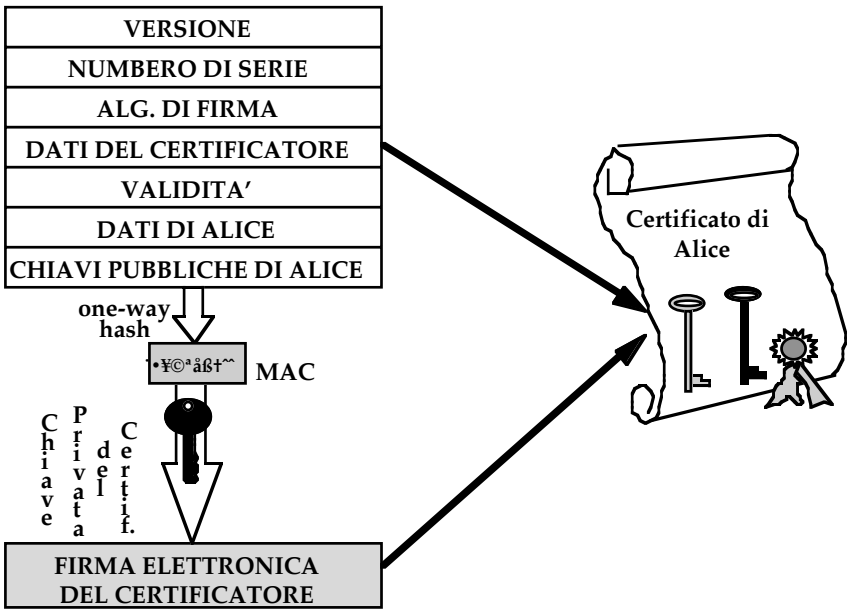

Figura 7. Formato di un certificato elettronico.

Distribuzione dei certificati

La firma elettronica dell'autorità garantisce sull'autenticità del certificato elettronico, e quindi della corrispondenza tra dati anagrafici e chiavi pubbliche.

E' però necessario che la *chiave pubblica dell'autorità sia nota a tutti,* per permettere a tutti di verificare l'autenticità di un certificato. Infatti chi riceve il certificato può derivare l'algoritmo di verifica da impiegare e l'identificativo dell'autorità dai campi che caratterizzano il certificato stesso. A questo punto, se egli conosce la chiave pubblica dell'autorità indicata nel certificato, egli può quindi verificare la firma elettronica inclusa, ovvero l'autenticità del certificato.

I certificati sottoscritti dall'autorità possono anche essere trasmessi in rete. La relazione che lega firma elettronica e corpo del certificato (attraverso l'uso di una funzione one-way hash) garantisce la non manomettibilità del certificato. Infatti, un piccolo cambiamento nel corpo del certificato darebbe luogo ad un MAC differente, e quindi necessiterebbe di una firma elettronica differente.

Si noti come il sistema dei certificati elettronici trasforma il problema dell'*acquisizione della chiave pubblica di un utente* nel problema dell'*acquisizione della sola singola chiave pubblica dell'autorità.*

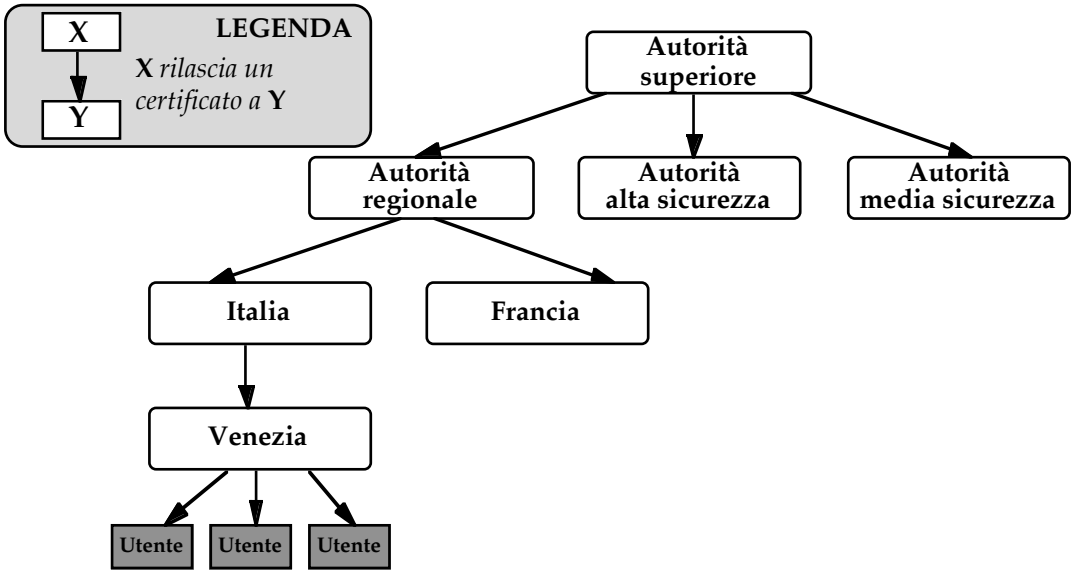

Figura 8. Albero delle autorità certificatrici.

Gerarchie di autorità per la certificazione

Quando il numero di utenti della rete cresce, diventa impossibile concentrare il compito del rilascio dei certificati su una singola *autorità certificatrice*. La soluzione è quella di creare una gerarchia di autorità, in modo da regionalizzare il sistema delle certificazioni. In questa gerarchia, rappresentabile appunto come un albero, ciascuna autorità certifica i propri figli. Le foglie sono gli utenti del servizio di certificazione, e ricevono il certificato dalle autorità al livello più basso della gerarchia.

In Figura 8 si illustra un esempio di albero gerarchico di autorità certificatrici. In particolare, sono rappresentati alcuni utenti che richiedono il certificato all'autorità veneziana.

Affinché questo sistema possa garantire la semplice autenticazione dei certificati, ogni utente deve conoscere a priori almeno la chiave dell'*Autorità superiore*, posta alla radice dell'albero. Infatti se un utente riceve un certificato firmato dall'autorità di *Venezia*, per essere sicuro dell'autenticità del certificato deve conoscere la chiave pubblica dell'autorità di *Venezia*, ovvero deve anche ricevere il certificato dell'autorita di *Venezia* rilasciato dall'autorità d'*Italia*. Questo processo di verifica può continuare fino a quando si riceve un certificato rilasciato dall'*autorità superiore*, la cui chiave pubblica è nota a priori.

Sistemi sicuri per la conservazione delle chiavi

Un sistema di sicurezza basato su un qualsiasi metodo di crittografia dipende esclusivamente dalla capacità dei singoli *utenti finali* di mantenere la *segretezza sulle chiavi*. I crittosistemi a chiave pubblica come RSA acuiscono questo problema, poiché richiedono chiavi talmente lunghe (*centinaia* di bit) da non poter essere ricordate a mente e da renderne difficoltosa la digitazione al terminale.

Le chiavi devono quindi essere mantenute su un opportuno *supporto di memoria*. Mentre l'impiego di un file per memorizzare le chiavi è ritenuto poco sicuro, il mezzo più sicuro è l'uso delle *smart card*, ovvero una scheda dotata di memoria non volatile che contiene la chiave codificata tramite un PIN (Personal Identifier Number). Per poter usare la chiave, l'utente deve solo digitare il PIN, che è un numero abbastanza piccolo da poter essere tenuto a mente. Per l'uso di smart card si rende necessario l'uso di un *lettore* di smart card, direttamente collegato al terminale o al calcolatore in rete

In conclusione, le *smart card* sono un mezzo abbastanza sicuro per conservare queste chiavi, a patto che la card sia usata in maniera strettamente personale, e che il lettore di card e il computer ad esso collegato siano protetti *fisicamente* da intrusioni esterne.

3. Esempio di transazione EDI sicura

Questa sezione presenta un esempio di comunicazione sicura tra il computer del mittente, Alice, ed il computer del destinatario, Bob. Alice e Bob sono i nomi dei due utenti tipici usati in letteratura per descrivere il processo usato per rendere sicure le comunicazioni.

Vedremo come tutti i concetti di crittografia simmetrica e asimmetrica, nonché di firma e certificato elettronico, sono impiegati per garantire tutte le proprietà di sicurezza di un MHS-EDI.

Questa tecnologia può essere applicata a qualsiasi sistema di messaggeria elettronica, non solo a messaggi in standard EDI. Tecnologie simili a quella descritta sono state infatti proposte per sistemi di posta elettronica quali lo standard CCITT/ISO X.435 (PEDI) e PEM (Privacy Enhanced Mail) di Internet. Un sistema di posta elettronica siffatto garantisce confidenzialità, autenticazione, non ripudiabilità per qualsiasi tipo di messaggi, e quindi anche per messaggi EDI.

I messaggi EDI possono però essere trasmessi anche attraverso l'uso di programmi di *file transfer* (come FTAM). Un *interchange* EDIFACT è infatti un file da trasmettere, che, tra i segmenti UNB e UNZ, racchiude un insieme di messaggi destinati allo stesso utente. Esistono già proposte di standard EDIFACT che prevedono l'aggiunta di ulteriori segmenti per garantire la sicurezza, come firme elettroniche, definizione degli algoritmi usati per la crittografia, certificati elettronici ecc. Lo schema di transazione sicura presentata in questa sezione si applica anche ai futuri standard di EDIFACT sicuro.

3.1. Confidenzialità e autenticazione del messaggio

Il requisito della confidenzialità nell'invio di un qualsiasi messaggio, e quindi anche di una transizione EDI, viene ottenuto attraverso una combinazione di schemi di crittografia, a chiave simmetrica e pubblica.

Il sistema a chiave pubblica è infatti troppo costoso per essere applicato a tutto il messaggio. Un algoritmo di crittografia/decrittografia a chiave pubblica diventa proponibile solo se si riduce la dimensione del dato a cui l'algoritmo stesso viene applicato.

La soluzione solitamente impiegata è basata su uno schema di crittografia a chiave simmetrica per crittografare tutto il messaggio. Il mittente (Alice) sceglie *casualmente* una chiave simmetrica K, detta *chiave di sessione*, usata per crittografare il messaggio. Per permettere al destinatario (Bob) di decrittografare il messaggio, Alice, oltre al messaggio crittografato con K, deve comunicare a Bob la chiave di sessione K. E' quindi importante trasmettere K in maniera confidenziale.

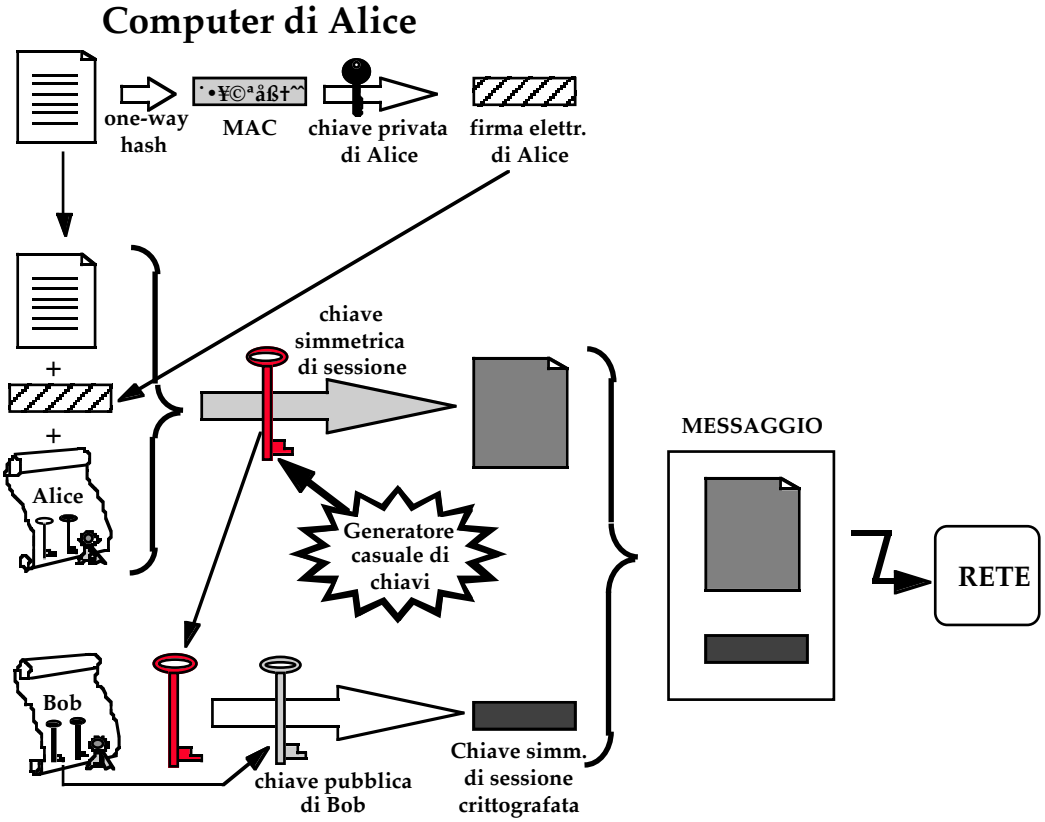

Figura 9. Invio di un messaggio con garanzie di confidenzialità e autenticazione.

Per garantire la trasmissione sicura di K, Alice crittografa K usando il metodo a chiave pubblica, ovvero crittografa usando la chiave pubblica K_p di Bob. Si noti che, poiché K è un'informazione molto più piccola dell'intero messaggio, diventa possibile applicare su K uno schema di crittografia a chiave pubblica. Si noti però che Alice deve conoscere K_p, la chiave pubblica di crittografia di Bob, ovvero Alice deve conoscere il certificato di Bob. In conclusione, usando questa combinazione di schemi di crittografia, il mittente di un messaggio deve conoscere il certificato del destinatario per garantire la confidenzialità del messaggio trasmesso.

L'autenticazione del messaggio viene garantito se il mittente (Alice) firma il elettronicamente il mesaggio inviato. La firma elettronica viene apposta da Alice impiegando la propria chiave privata. Il destinatario (Bob) deve conoscere la corrispondente chiave pubblica di Alice per poter verificare l'autenticità del messaggio ricevuto, ovvero Bob deve conoscere il certificato di Alice per poter accedere alla sua chiave pubblica. In conclusione, il destinatario di un messaggio deve conoscere il certificato del mittente per poter verificare l'autenticità del messaggio stesso.

Nella sezione seguente vedremo come Alice possa direttamente comunicare a Bob il proprio certificato assieme al messaggio.

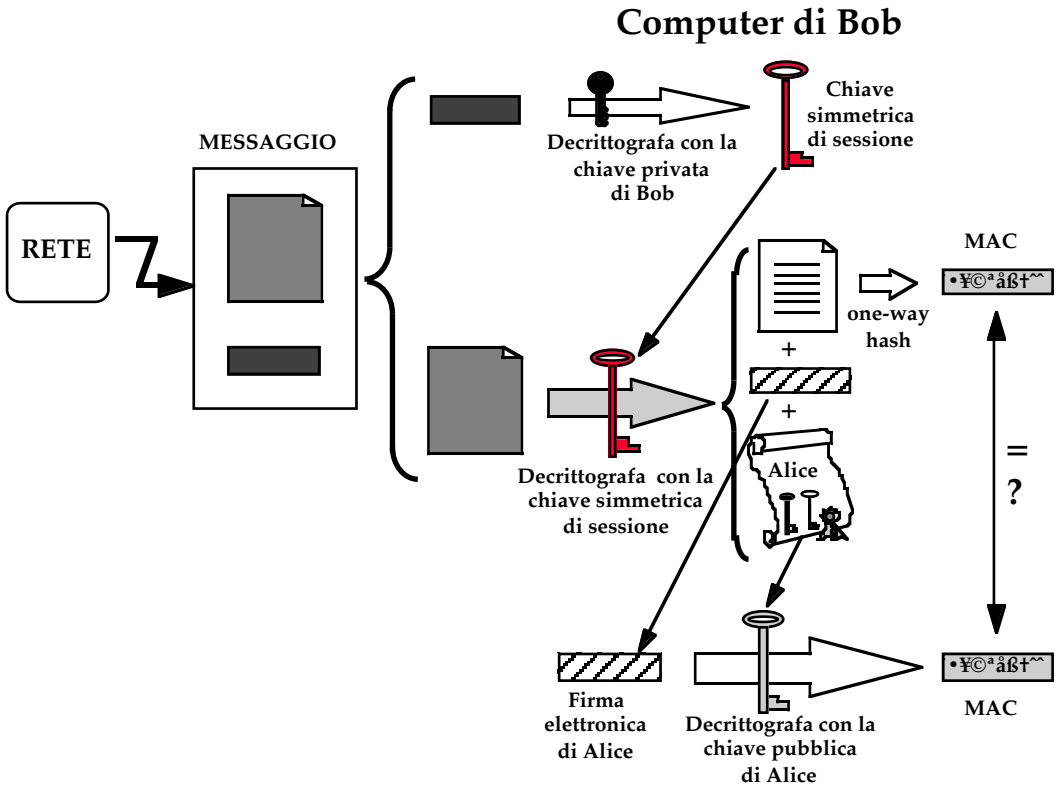

Figura 10. Ricezione di un messaggio con garanzie di confidenzialità e autenticazione.

3.2. Schema di transazione sicura

In questa sezione si illustrerà uno schema di trasmissione sicura, che garantisca dal punto di vista della confidenzialità del messaggio trasmesso e dell'autenticazione del mittente.

In Figura 9 si illustra lo schema di invio di un messaggio da parte del mittente della trasmissione (Alice). Si noti la generazione della chiave simmetrica di sessione e la firma elettronica del messaggio. Si noti ancora come Alice può trasmettere il suo certificato assieme al messaggio stesso. Ciò semplifica molto il primo contatto tra Alice ed un nuovo partner. Affinché lo schema funzioni, però, Alice deve però conoscere il certificato di Bob prima di iniziare la trsmissione. Ad esempio, Alice potrebbe richiedere preventivamente il certificato di Bob con una comunicazione non sicura dal punto di vista della confidenzialità. Si ricordi che il meccanismo di firma elettronica dei certificati li protegge anche da manomissioni.

In Figura 10 si illustra lo schema di ricezione di un messaggio da parte del destinatario della trasmissione (Bob). Il processo inizia decrittografando la chiave simmetrica di sessione, dopodiché la chiave pubblica di Alice contenuta nel suo certificato viene usata per verificare l'autenticità della firma elettronica.

Si noti che Bob, prima di impiegare la chiave pubblica di Alice estraendola dal certificato, dovrebbe prima verificare l'autenticità del certificato stesso. La verifica va effettuata controllando la firma elettronica apposta dall'autorità sul certificato, il che richiede però la conoscenza della chiave pubblica dell'autorità. Questa verifica preventiva del certificato ricevuto non viene illustrata in Figura 10.

3.3. Non ripudiabilità di invio e ricezione

Una volta che il destinatario ha ricevuto un messaggio, la firma elettronica apposta dal mittente garantisce non solo sull'*autenticità* del messaggio, ma anche sulla *non ripudiabilità* dell'invio del messaggio stesso.

Infatti un mittente non può ripudiare un messaggio inviato e firmato elettronicamente poiché solo quel mittente può essere a conoscenza della *chiave privata* usata per la firma elettronica. Una proprietà importante del meccanismo di firma elettronica è la possibilità di coinvolgere terze parti per la verifica della validità della firma stessa. La terza parte verifica usando la chiave pubblica del mittente del messaggio, e non ha bisogno di conoscere la sua chiave privata.

La *non ripudiabilità* della ricezione è un altro requisito garantito tramite la firma, ma necessita che il MHS invii al mittente un messaggio di *ricevuta di ritorno* nel momento in cui il destinatario preleva un messaggio dalla propria Message Store. Se il messaggio di ritorno viene firmato elettronicamente tramite la chiave privata del destinatario, questo garantisce la non ripudiabilità della ricezione. Naturalmente un identificativo univoco del messaggio deve essere contenuto nel messaggio di ritorno, ad esempio la data di invio del messaggio e il numero di sequenza del messaggio stesso.

3.4. Integrità del messaggio e della sequenza dei messaggi

Il meccanismo usato per la firma elettronica garantisce anche sull'*integrità del messaggio*. Si condsideri che la firma elettronica è ottenuta attraverso una funzione *one-way hash* che produce un MAC. Se il messaggio ricevuto non è integro (in qualcuna delle sue parti), il controllo della firma elettronica presso il destinario *fallisce.*

Inoltre i messaggi non possono essere persi o duplicati se l'MHS trasmette una *marca* progressiva che tiene conto della sequenza dei messaggi tra ogni coppia mittente/destinatario. La funzione *one-way hash* deve essere applicata a *corpo del messaggio + marca* per produrre il MAC per la firma elettronica. A questo punto se il messaggio è ricevuto integro, il destinatario può controllare dalla marca eventuali perdite o duplicazioni di messaggi

4. Iniziative correnti per la sicurezza del commercio elettronico

Il tema della sicurezza sta diventando sempre più importante, soprattutto nel momento in cui le reti di calcolatori sono usate per il commercio elettronico.

Proposte essenzialmente simili sono formulate all'interno di vari gruppi e organizzazioni. Nell'ambito dello standard ISO/OSI X.400 sono state proposte molte delle tecnologie per la sicurezza attualmente adottate da tutte le altre proposte/realizzazioni successive. Ad esempio, X.500 per il naming univoco dei soggetti in rete, e X.509, il formato standard per i certificati elettronici basato su X.500.

Sotto gli auspici della Internet Architecture Board, è stato recentente proposto un sistema di posta *sicuro* detto Privacy Enhanced Mail (PEM). La Netscape Inc., la società diventata famosa per il browser WEB, ha fissato una tecnologia per comunicare in maniera sicura con server commerciali. La Microsoft Corp. e la Visa hanno stipulato un accordo per stabilire uno standard di sicurezza per le transazioni finanziarie (Electronic Cash). Infine, all'interno dei gruppi che si occupano della standardizzazione EDIFACT, è stato proposto uno standard di messaggio che include le solite caratteristiche di sicurezza.

4.1. ISO/OSI (X.400)

Lo standard CCITT/ISO X.400 è un livello *applicazione* che definisce un servizio di posta elettronica a *formato libero* costruito sui livelli inferiori dello standard ISO/OSI.

Nell'ambito di questo standard, X.500 determina il servizio di naming dei soggetti in rete (servizio di directory elettronica), e X.509 stabilisce il formato dei certificati usando la sintassi X.500 per i dati identificativi. X.500 e X.509 sono strettamente collegati, in quanto i dati identificativi dell'utente definiti da X.500 sono quelli da includere nei certificati standardizzati da X.509.

Sempre nell'ambito X.400, la specifica X.435 definisce le modalità *sicure* per il trasporto di messaggistica EDI su X.400. X.435, detto PEDI, definisce i livelli di sicurezza e le modalità per ottenere tali requisiti. Poiché in questo caso il messaggio non è più a formato libero, viene definita una strutturazione del corpo del messaggio, con intestazioni varie (header) che trasportano le informazioni necessarie per implementare i meccanismi di sicurezza (definizioni di algoritmi di crittografia impiegati, certificati, ecc.) illustrati precedentemente. I requisiti di sicurezza sono infatti raggiunti tramite combinazione di crittografia a chiave simmetrica o pubblica.

4.2. Internet (servizio di posta elettronica)

Nell'ambito Internet, è stato proposto uno standard per la trasmissione tramite SMTP (che è lo standard de-facto per la posta elettronica) di messaggi elettronici con dei livelli di sicurezza. Il suo acronimo è PEM (Privacy Enhanced Mail).

PEM non è stato progettato specificatamente per EDI, ma realizza i requisiti di sicurezza per qualsiasi tipo di messaggio (formato libero).

I requisiti di sicurezza sono ancora raggiunti tramite combinazione di crittografia a chiave simmetrica o pubblica. Il meccanismo dei certificati elettronici è definito sulla base degli standard X.500 e X.509.

4.3. Il server commerciale sicuro di Netscape

La Netscape Communications Corporation, la ditta che sviluppa il famoso browser di rete, ha sviluppato una tecnologia che permette di comunicare in maniera sicura nei confronti di un *Server* su Internet. Questa tecnologia garantisce:

- l'*autenticazione* del *Server* nei confronti di qualsiasi cliente che accede ad esso;
- la *confidenzialità* della comunicazione tra cliente e *Server*;
- la garanzia dell'*integrità* dei messaggi trasmessi.

In altri termini, Netscape ha sviluppato dei server di rete WWW sicuri (tecnicamente, server che adottano un protocollo sicuro chiamato https). Questo protocollo permette la privatezza delle comunicazioni con il server, rendendo possibili, ad esempio, transazioni finanziarie (es. acquisti) attraverso la comunicazione *confidenziale* del numero di carta di credito. Inoltre, permette al cliente di verificare l'autenticità del server, grazie all'impiego di un certificato elettronico associato al server.

Si noti però che questa tecnologia non garantisce l'autenticazione del cliente nei confronti del server. Essa è infatti stata pensata per applicazioni commerciali, dove il cliente spende dei soldi, e deve essere sicuro che il server, ovvero il negozio da cui si sta servendo, non sia un impostore che vuole solo rapinare del denaro e non vendere dei prodotti.

La tecnologia è basata su un crittosistema a chiave pubblica RSA, con chiavi abbastanza corte (40 bit) fuori dagli U.S.A. (perché solo questa tecnologia è esportabile), e con chiavi più lunghe (128 bit) in U.S.A.
I certificati (per i Server) sono rilasciati dall'RSA Certificate Service, un servizio dell' RSA Data Security, Inc.

Si noti che ogni browser Netscape ha conoscenza della chiave pubblica del certificatore per verificare l'autenticità dei certificati. Tale chiave è built-in in ogni browser.

4.4. Elettronic cash (Visa e Microsoft)

Visa International e Microsoft Corportion stanno definendo una tecnologia per rendere sicure le transazioni bancarie come pagamenti tramite *carta di credito* o *bonifico*. Lo standard tecnologico, noto come STT (Secure Transaction Technology), è stato definito su *reti aperte*. Lo standard è stato reso pubblico per permettere a venditori di software di sviluppare applicazioni che possano interoperare con altre implementazioni di STT. Recentemente molti altri partner industriali si sono uniti a Visa International e Microsoft Corportion per partecipare alla definizione dello standard.

La tecnologia di base su cui STT è stata definita è sempre la stessa, e prevede un uso combinato di crittografia a chiave simmetrica e pubblica.

4.5. Sicurezza nello standard EDIFACT

Nell'ambito dei comitati delle Nazioni Unite per la standardizzazione del commercio elettronico sono in discussione standard di messaggio UN/EDIFACT con caratteristiche di sicurezza. In particolare, lo standard dovrà realizzare tutti i requisiti di sicurezza discussi precedentemente.

Una delle proposte è basata sull'adozione di uno standard di messaggio particolare, che introduce speciali *segmenti di sicurezza* (*header* e *trailer* del messaggio vero e proprio). Gli header specificano i metodi di sicurezza applicati al messaggio stesso (es. gli algoritmi usati per la crittografia e le funziioni one-way hash). I trailer contengono i risultati dell'applicazione dei metodi di sicurezza al messaggio stesso (es. firma elettronica).

La proposta nell'ambito UN/EDIFACT adotta ancora le stesse tecnologie descritte precedentemente, ma propone cambiamenti significativi del formato dei messaggi EDIFACT. La proposta UN/EDIFACT sposta al livello più alto della gerarchia dei protocolli le problematiche di sicurezza.

5. Riferimenti

[1] W.R. Cheswick, S.M. Bellovin. *Firewalls and Internet Security.* Addison-Wesley, 1994.

[2] S. Garfinkel. *PGP: Pretty Good Privacy.* O'Really & Associates, Inc., 1995.

[3] S.T. Kent, "Internet Privacy Enhanced Mail", *Comm. of ACM*, Vol. 36, No. 8, pp. 48-59, August 1993.

[4] Microsoft Corporation and Visa International. "Secure Transaction Tecnology Specifications", 1995.

[5] Verisign, Inc. "Introduction to Cryptography", 1995.

[6] N.J. Yeager, R.E. McGrath, *Web Server Technology.* Morgan Kaufmann Publisher, Inc., 1996.

[7] "Cryptography Frequently Asked Questions", 1993, `ftp://pub/usenet/news.answers/cryptography-faq/part[xx]`.

[8] "Security Frequently Asked Questions", 1993.

[9] "Frequently Asked Questions about Electronic Data Interchange (EDI) on the Internet", 1995, `ftp://ds.internic.net/internet-drafts/draft-ietf-edi-faq-01.txt`.

[10] "Security for UN/EDIFACT Message Transfer", 1993, Draft submitted by the Rapporteur for the West European Board

[11] "Security for UN/EDIFACT Message Transfer", 1993, Draft submitted by the Rapporteur for the West European Board