3 CONCEPTE ȘI ALGORITMI DE SECURITATE RFD

3.1 Confidențialitatea mesajelor
3.2 Integritatea datelor
3.3 Identificarea participanților
3.4 Identificarea biometrică
3.5 Autentificarea participanților
3.6 Controlul accesului
3.7 Refuzuri/întreruperi în prestarea serviciului
3.8 Nonrepudierea
3.9 Gestionarea cheilor cifrografice
3.10 Schimbul cheilor secrete
3.11 Gestionarea certificatelor
3.12 Autentificarea părților
3.13 Securitatea la spargeri
3.14 PGP, OpenPGP și GnuPG

3.1 Confidențialitatea mesajelor

3.1.1 Cifrografia simetrică

Confidențialitatea garantează că informațiile vor fi comunicate doar părților care sunt autorizate să le primească.

Tăinuirea este realizată cu ajutorul algoritmilor de cifrare.

Există două tipuri de cifrare: cifrarea simetrică, în care operațiile de cifrare și de descifrare a mesajului utilizează aceeași cheie secretă și cifrarea cu chei publice, unde atât cifrarea cât și descifrarea se efectuează cu o cheie secretă și una publică, iar în cazul de semnături numerice – cifrarea (semnarea) se efectuează cu o cheie secretă, iar descifrarea (confirmarea semnăturii) cu o cheie publică.

Fie M mesajul care urmează să fie cifrat cu o cheie simetrică K în procesul de cifrare E. Rezultatul va fi textul cifrat C astfel încât

$$E[K(M)] = C.$$

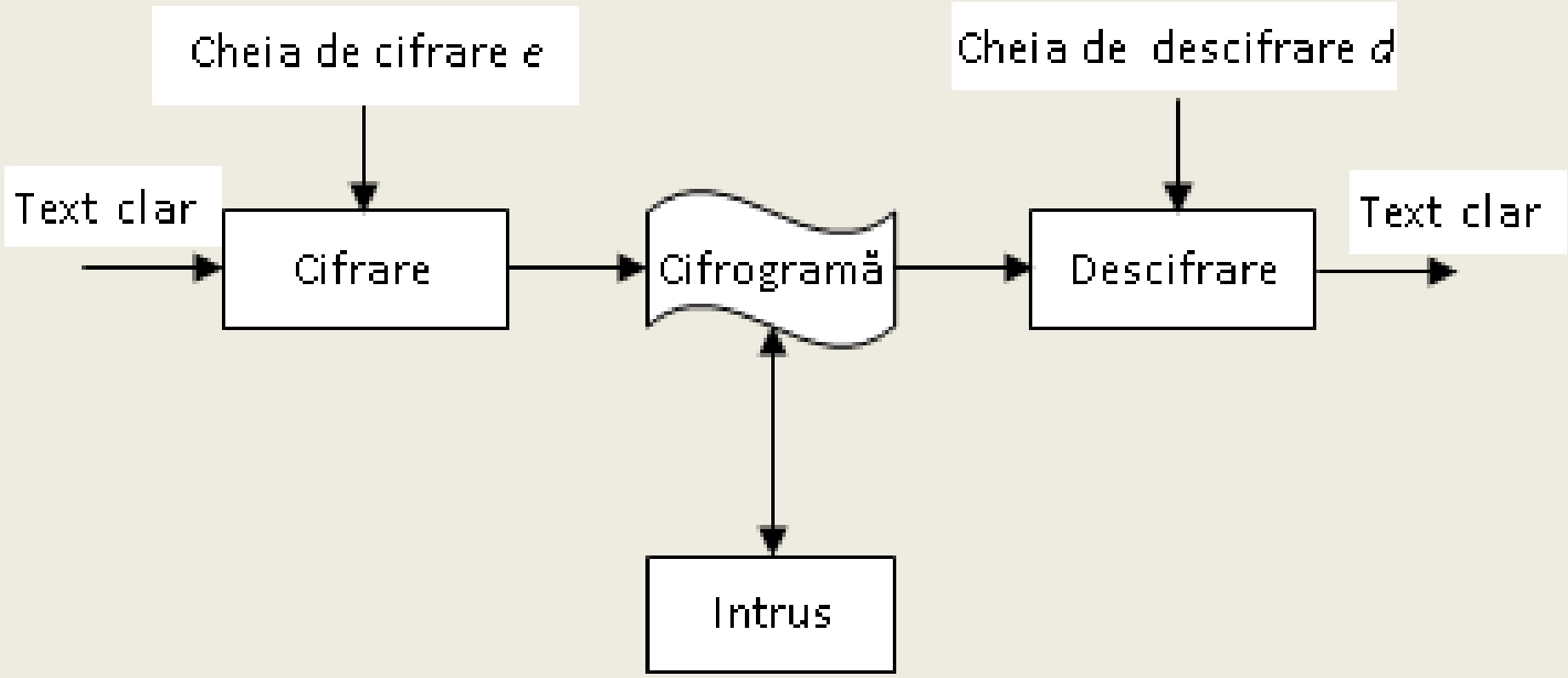
Procesul de descifrare D este funcția inversă a lui E care restabilește textul clar:

$$D[K(C)] = M.$$

3.1 Confidențialitatea mesajelor

3.1.1 Cifrografia simetrică

Schema unui sistem cifrografic simetric *(d = e)*:



3.1 Confidențialitatea mesajelor

Algoritmi de cifrare simetrică în i-comerț [3]:

Algorithms	Name and Description	Block Size in Bits	Key Length in Bits	Standard
AES	Advanced Encryption Standard	Blocks of 128, 192, or 256 bits	128, 192, or 256	FIPS 197
DES	Data Encryption Standard	Blocks of 64 bits	56	FIPS 81, ANSI X3.92, X3.105, X3.106, ISO 8372, ISO/IEC 10116
IDEA (Lai and Massey, 1991a,b)	International Data Encryption Algorithm	Blocks of 64 bits	128	—
RC2	Developed by Ronald Rivest (Schneier, 1996, pp. 319–320)	Blocks of 64 bits	Variable (previously limited to 40 bits for export from the United States)	No; proprietary
RC4	Developed by R. Rivest (Schneier, 1996, pp. 397–398)	Stream	40 or 128	No, but posted on the Internet in 1994
RC5	Developed by R. Rivest (1995)	Blocks of 32, 64, or 128 bits	Variable up to 2048 bits	No; proprietary
SKIPJACK	Developed for applications with the PCMCIA card Fortezza	Blocks of 64 bits	80	Declassified algorithm; version 2.0 available at http://csrc.nist.gov/groups/ST/toolkit/documents/skipjack/skipjack.pdf, last accessed January 25, 2016
Triple DES	Also called TDEA	Blocks of 64 bits	112	ANSI X9.52/NIST SP 800-67 (National Institute of Standards and Technology, 2012a)

Note: FIPS, Federal Information Processing Standard.

3.1 Confidențialitatea mesajelor

3.1.1 Cifrografia simetrică

Principalul dezavantaj al sistemelor de cifrare simetrică este că ambele părți trebuie să obțină, într-un fel sau altul, cheia unică de cifrare. Acest lucru este posibil fără prea multe probleme în cadrul unei organizații închise; pe rețele deschise, schimbul poate fi interceptat.

Cifrografia cu chei publice, propusă în 1976 de către Diffie și Hellman, este o soluție la problema schimbului de chei. ♦

4.6 Confidențialitatea mesajelor

3.1.2 Cifrografia cu chei publice

Cifrografia cu chei publice, propusă în 1976 de către Whitfield **Diffie și** Martin **Hellman, este o soluție la problema schimbului de chei pentru sistemele cifrografice simetrice. Ulterior a devenit cunoscut că sistemul cu chei publice a fost propus în 1969 în cadrul Serviciul Secret al Marii Britanii, dar nu a fost publicat, fiind considerat secret militar.**

Sistemul propus prevede folosirea unui sistem criptografic asimetric. La *sistemele criptografice asimetrice* cheia de descifrare este diferită de cea de cifrare şi nu poate fi, practic, dedusă din aceasta. Pentru ca un sistem criptografic asimetric să asigure atât confidențialitatea, cât şi integritatea informaţiei, se cere ca algoritmul de cifrare *E* şi cel de descifrare *D* să satisfacă trei cerinţe:

***1) D*(*E*(*M*)) = *E*(*D*(*M*)) = *M*. Aici *M* este textul clar, care trebuie cifrat şi transmis;**

2) este mai mult decât dificil a se deduce *D* din *E*;

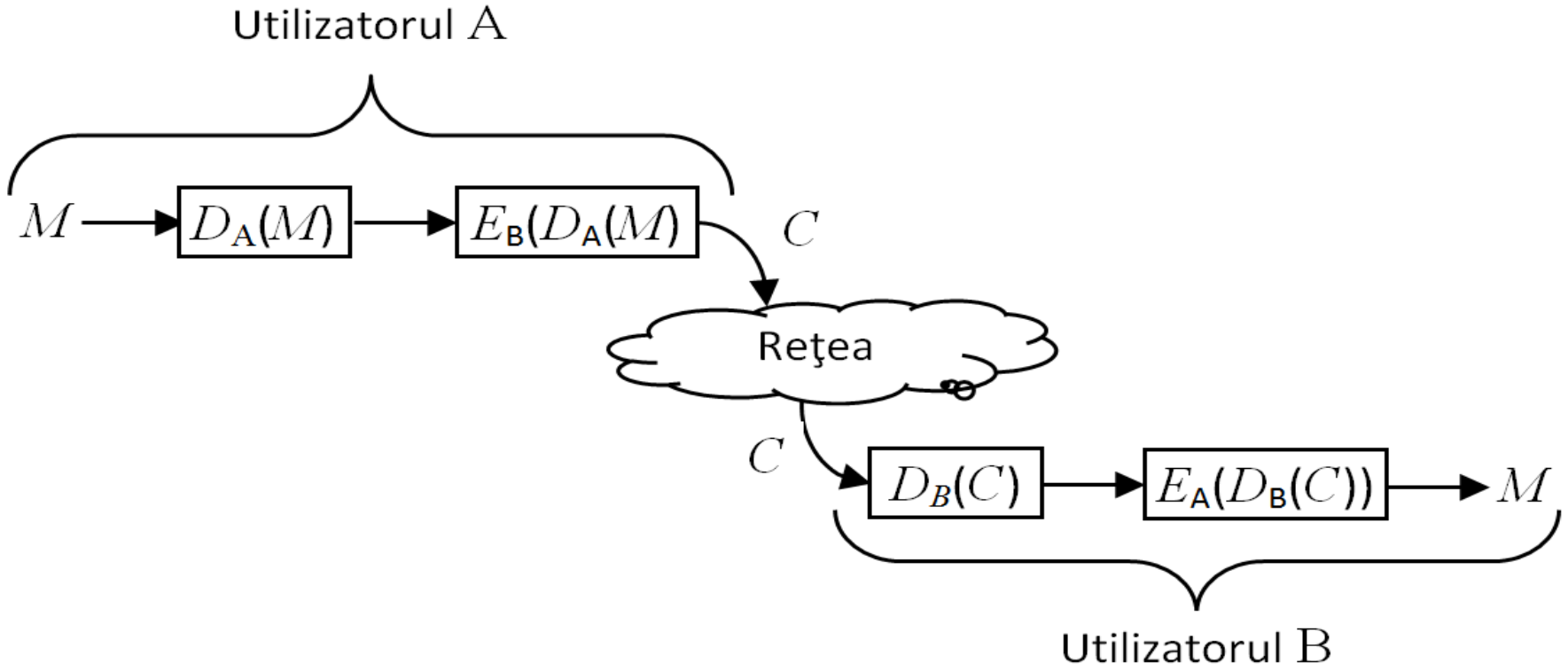
***3) E* nu poate fi spart prin criptanaliză cu text clar ales.**

O cheie se face cunoscută tuturor – *cheie publică*, pe când o altă cheie se ţine în secret – *cheie privată*.

3.1 Confidențialitatea mesajelor

3.1.2 Cifrografia cu chei publice

Modelul sistemului criptografic cu chei publice:



3.1 Confidențialitatea mesajelor

3.1.2 Cifrografia cu chei publice

Sunt bine cunoscuţi asemenea algoritmi pentru sistemele criptografice cu chei publice ca:

- **algoritmul RSA propus de Ron Rivest, Adi Shamir şi Leonard Adleman în 1978. Securitatea acestuia se bazează pe dificultatea factorizării numerelor mari;**
- **algoritmul propus de Ralph Merkle în 1978. Securitatea acestuia se bazează pe dificultatea determinării conţinutului unui rucsac cunoscând greutatea (volumul) lui;**
- **algoritmul propus de Taher ElGamal în 1985 şi cel propus de Schnorr în 1991 se bazează pe dificultatea calculului logaritmilor discreţi;**
- **algoritmul bazat pe curbe eliptice propus de Alfred J. Menezes şi Scott A. Vanstone în 1993;**
- **algoritmul DSA (Digital Signature Algorithm) propus de David Kravitz în 1991.**

3.1 Confidențialitatea mesajelor

3.1.2 Cifrografia cu chei publice

Standardul de facto pentru cifrarea cu chei publice este algoritmul RSA elaborat de Rivest et al. (1978). Cu toate acestea, în multe aplicații noi, cifrografia curbelor eliptice (*elliptic curve cryptography* – ECC) oferă avantaje semnificative

Compararea RSA și ECC după lungimea cheilor necesare pentru a asigura același nivel de securitate [3]:

RSA	Elliptic Curve	Reduction Factor RSA/ECC
512	106	5:1
1,024	160	7:1
2,048	211	10:1
5,120	320	16:1
21,000	600	35:1

3.1 Confidențialitatea mesajelor

3.1.2 Cifrografia cu chei publice

Recomandările NIST (2012) privind lungimea cheilor în biți:

Symmetric	RSA and Diffie–Hellman	Elliptic Curve
80	1,024	160–112
112	2,048	224–255
128	3,072	256–283
192	7,680	384–511
256	15,360	512+

♦

3.2 Integritatea datelor

Obiectivul serviciului de integritate este eliminarea tuturor posibilităților de modificare neautorizată a mesajelor în timpul tranzitului de la expeditor la receptor.

Forma tradițională pentru realizarea acestei securități este ștampilarea plicului scris cu sigiliul de ceară al expeditorului.

Transpunerea acestui concept la i-tranzacții, sigiliul va fi o secvență de biți asociate în mod univoc cu documentul care trebuie protejat.

Această secvență de biți va constitui o "amprentă numerică" unică și nefalsificabilă care va însoți documentul trimis la destinație.

Receptorul va recalcula valoarea amprentei din documentul primit și va compara valoarea obținută cu valoarea trimisă.

Orice diferență va indica faptul că integritatea mesajului a fost încălcată.

Amprenta poate fi făcută dependentă de conținutul mesajului numai prin aplicarea unei funcții hash (de amestec).

O funcție hash convertește o secvență de caractere de orice lungime într-un lanț de caractere cu o lungime fixă, L, de obicei mai mică decât lungimea textului, numită valoare hash.

3.2 Integritatea datelor

Valoarea hash are numeroase nume: compresie, contracție, mesaj rezumat, amprentă numerică, sumă de control cifrografică, de verificare a integrității mesajului (*message integrity check* - MIC) ș.a.

Dacă algoritmul hash este cunoscut, orice entitate poate calcula valoarea hash din mesaj folosind funcția hash.

Din motive de securitate, valoarea hash depinde de conținutul mesajului și de cheia privată a expeditorului, în cazul unui algoritm de cifrare cu chei publice, sau al unei chei secrete pe care numai expeditorul și receptorul o cunosc, în cazul unui algoritm de cifrare simetric.

În primul caz, oricine cunoaște funcția hash poate calcula amprenta cu cheia publică a expeditorului.

În cel de-al doilea caz, numai receptorul dorit va fi capabil să verifice integritatea.

Trebuie remarcat faptul că lipsa de integritate poate fi folosită pentru a sparge confidențialitatea.

De exemplu, confidențialitatea unor algoritmi poate fi spartă prin atacuri asupra vectorilor de inițializare. ♦

3.2 Integritatea datelor

Algoritmii hash utilizați în mod obișnuit în i-comerț [3]:

Algorithm	Name	Signature Length (*L*) in Bits	Block Size (*B*) in Bits	Standardization
AR/DFP	Hashing algorithms of German banks			German Banking Standards
DSMR	Digital signature scheme giving message recovery			ISO/IEC 9796
MCCP	Banking key management by means of public key algorithms using the RSA cryptosystem; signature construction by means of a separate signature			ISO/IEC 1116-2
MD4	Message digest algorithm	128	512	No, but described in RFC 1320
MD5	Message digest algorithm	128	512	No, but described in RFC 1321
NVB7.1, NVBAK,	Hashing functions used by Dutch banks			Dutch Banking Standard, published in 1992
RIPEMD	Extension of MD4, developed during the European project RIPE (Menezes et al., 1997, p. 380)	128	512	
RIPEMD-128	Dedicated hash function #2	128	512	ISO/IEC 10118-3
RIPEMD-160	Improved version of RIPEMD (Dobbertin et al., 1996)	160	512	
SHA	Secure Hash Algorithm (replaced by SHA-1)	160	512	FIPS 180
SHA1 (SHA-1)	Dedicated Hash Function #3 (revision and correction of the Secure Hash Algorithm)	160	512	ISO/IEC 10118-3
				FIPS 180-1 (National Institute of Standards and Technology, 1995), FIPS 180-4 (National Institute of Standards and Technology, 2012b)
SHA-2		224, 256	512	FIPS 180-4
		384, 512	1024	

Compararea unor algoritmi cifrografici de hashare (amprentă):

Algorithm and variant		Output size (bits)	Internal state size (bits)	Block size (bits)	Rounds	Operations	Security (in bits) against collision attacks	Capacity against length extension attacks	Performance on Skylake (median cpb)[55]		First published
									long messages	8 bytes	
MD5 (as reference)		128	128 (4 × 32)	512	64	And, Xor, Rot, Add (mod 2^{32}), Or	≤18 (collisions found)[56]	0	4.99	55.00	1992
SHA-0		160	160 (5 × 32)	512	80	And, Xor, Rot, Add (mod 2^{32}), Or	<34 (collisions found)	0	≈ SHA-1	≈ SHA-1	1993
SHA-1							<63 (collisions found[57])		3.47	52.00	1995
SHA-2	*SHA-224*	224	256 (8 × 32)	512	64	And, Xor, Rot, Add (mod 2^{32}), Or, Shr	112	32	7.62	84.50	2004
	SHA-256	256					128	0	7.63	85.25	2001
	SHA-384	384	512 (8 × 64)	1024	80	And, Xor, Rot, Add (mod 2^{64}), Or, Shr	192	128 (≤ 384)	5.12	135.75	2001
	SHA-512	512					256	0	5.06	135.50	
	SHA-512/224	224					112	288	≈ SHA-384	≈ SHA-384	2012
	SHA-512/256	256					128	256			
SHA-3	*SHA3-224*	224	1600 (5 × 5 × 64)	1152	24[58]	And, Xor, Rot, Not	112	448	8.12	154.25	2015
	SHA3-256	256		1088			128	512	8.59	155.50	
	SHA3-384	384		832			192	768	11.06	164.00	
	SHA3-512	512		576			256	1024	15.88	164.00	
	SHAKE128	*d* (arbitrary)		1344			min(*d*/2, 128)	256	7.08	155.25	
	SHAKE256	*d* (arbitrary)		1088			min(*d*/2, 256)	512	8.59	155.50	

3.2 Integritatea datelor

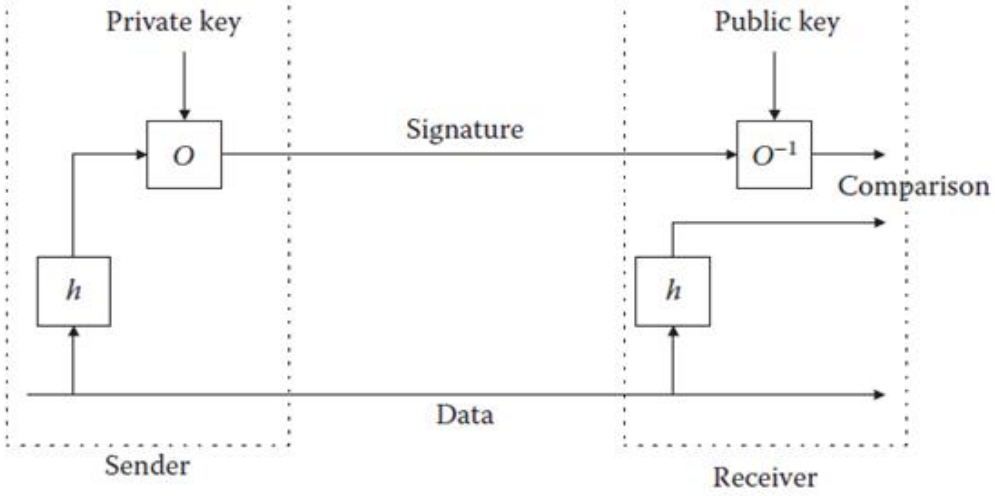
Verificarea integrității cu cifrografia cu chei publice

În cazul folosirii unui algoritm cu chei publice, un element de informație M care este cifrat de cheia privată SK_X a unei entități *X* poate fi citit de orice utilizator care posedă cheia publică corespunzătoare PK_X. Un expeditor poate, prin urmare, să semneze un document prin cifrarea acestuia cu o cheie privată rezervată operațiunii de semnătură pentru a produce sigiliul care însoțește mesajul. Orice persoană care cunoaște cheia publică corespunzătoare va putea să descifreze sigiliul și să verifice dacă acesta corespunde mesajului primit.

Un alt mod de a produce semnătura folosind cifrografia cu chei publice este de a cifra amprenta documentului. Acest lucru se datorează faptului că cifrarea unui document de dimensiune mare folosind un algoritm cu chei publice impune calcule substanțiale și introduce întârzieri excesive. De aceea este benefic de utilizat un rezumat (*digest*) al mesajului inițial înainte de a aplica cifrarea. Aceast rezumat este produs prin aplicarea unei funcții de hashare unică pentru a calcula amprenta care este apoi cifrată cu cheia privată a expeditorului. La destinație, receptorul recalculează amprenta. Cu cheia publică a expeditorului, receptorul va putea descifra amprenta pentru a verifica dacă valoarea hash primită este identică cu valoarea hash calculată. Dacă ambele sunt identice, semnătura este validă.

Verificarea integrității cu cifrografia cu chei publice

Verificarea integrității folosind cifrarea cu chei publice a rezumatului *hash* al mesajului [3]:



Algoritmi cu chei publice, frecvent utilizați pentru calcularea semnăturilor numerice:

Algorithm	Comments	Signature Length in Bits	Standard
DSA	Digital Signature Algorithm is a variant of the ElGamal algorithm and published the Digital Signature Standard (DSS) proposed by NIST (National Institute of Standards and Technology) in 1994.	320, 448, 512	FIPS 186-4 (National Institute of Standards and Technology, 2013)
ECDSA	Elliptic Curve Digital Signature Algorithm, first standardized in 1998.	384, 488, 512, 768, 1024	ANSI X9.62:2005
RSA	This is the *de facto* standard algorithm for public key encryption; it can also be used to calculate signatures.	512–1024	ISO/IEC 9796

Semnătură oarbă

O semnătură oarbă este o procedură specială pentru ca un notar să semneze un mesaj folosind algoritmul RSA pentru cifrografia cu chei publice fără a dezvălui conținutul. O posibilă utilizare a acestei tehnici este de a notifica timpul i-plăților.

Fie un debitor care ar dori să aibă o plată semnată orb de o bancă. Banca are o cheie publică *e*, o cheie privată *d* și un modulo public *N*. Debitorul alege un număr aleator *k* între 1 și *N* și păstrează acest număr secret.

Plata *p* este "încapsulată" prin aplicarea (pk^e) mod *N*, înainte de transmiterea mesajului către bancă. Banca semnează mesajul cu cheia privată, adică

$$(pk^e)^d \bmod N = p^d k \bmod N$$

și returnează plata debitorului. Debitorul poate extrage nota semnată împărțind numărul la *k*. Pentru a verifica dacă nota primită de la bancă este cea care a fost trimisă, debitorul o poate ridica la puterea *e* deoarece

$$(p^d)^e \bmod N = p \bmod N.$$

Diferitele protocoale de plată pentru i-bani profită de semnăturile oarbe pentru a satisface condițiile de anonimat. ♦

Verificarea integrității cu cifrografia simetrică

Codul de autentificare a mesajelor (MAC, amprenta, valoarea verificării integrității) este rezultatul unei funcții hash unidirecționale care depinde de o cheie secretă.

Acest mecanism garantează simultan integritatea conținutului mesajului și autentificarea expeditorului.

Modul cel mai evident de a construi un MAC, este de a cifra valoarea hash-ului cu un algoritm de cifrare bloc simetric.

MAC-ul este apoi atașat la mesajul inițial, iar întregul obținut este trimis către receptor.

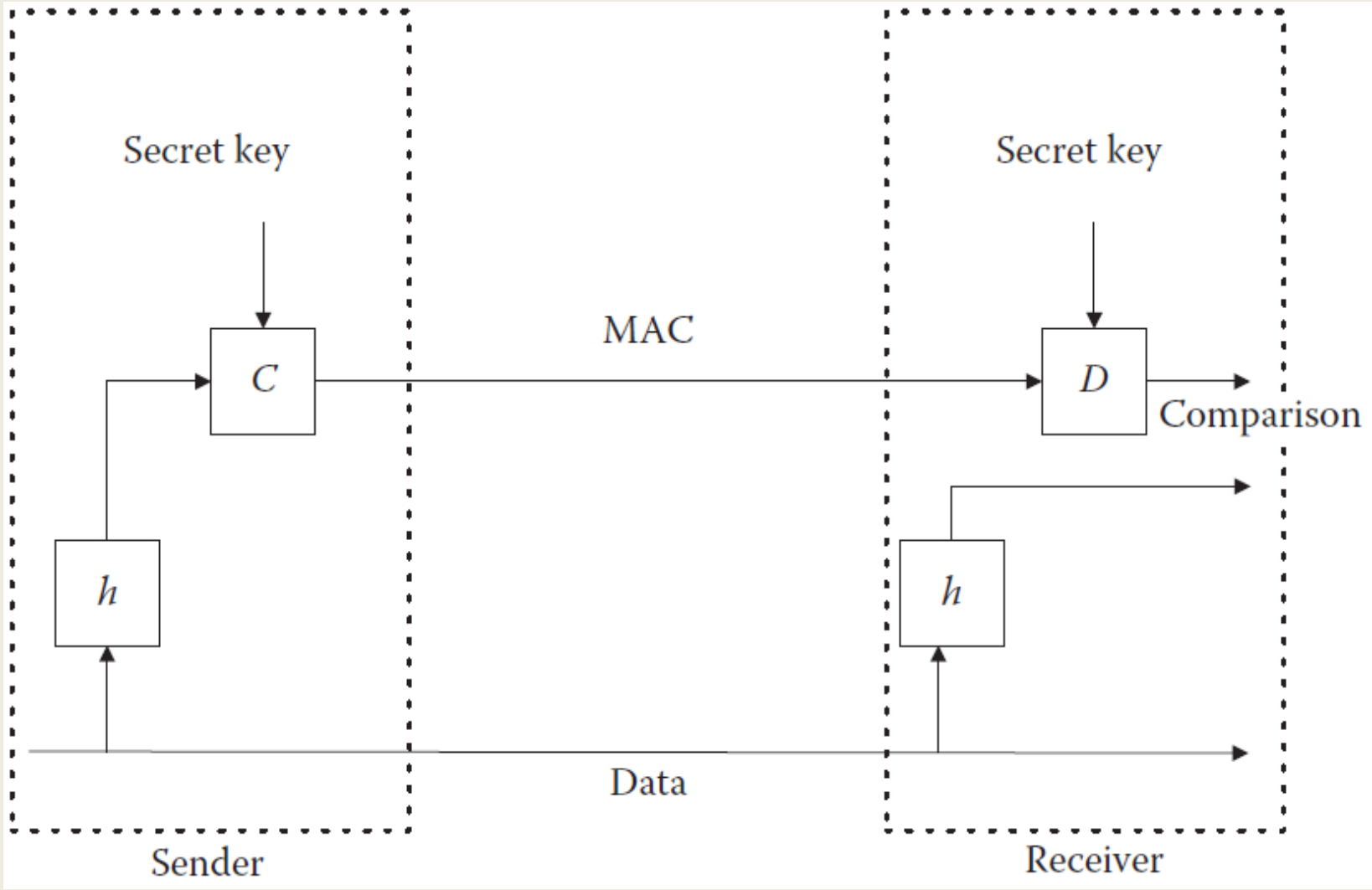
Receptorul recalculează valoarea hash, aplicând aceeași funcție hash asupra mesajului primit, și compară rezultatul obținut cu valoarea MAC descifrată.

Egalitatea ambelor rezultate confirmă integritatea datelor.

O altă variantă a acestei metode este de a atașa cheia secretă la mesajul ce va fi condensat cu funcția hash.

Verificarea integrității cu cifrografia simetrică

Operațiile de verificare în care *h* reprezintă funcția hash, *C* funcția de cifrare și *D* funcția de descifrare.



3.3 Identificarea participanților

Identificarea este procesul de constatare a identității unui participant (fie că este vorba de o persoană sau de un echipament) în baza unei caracteristici unice distinctive. Aceasta contrastează cu autentificarea, care este confirmarea faptului că identificatorul distinctiv corespunde cu adevărat utilizatorului declarat.

Autentificarea și identificarea unei entități care comunică are loc simultan atunci când una dintre părți trimite celuilalt în mod privat un secret care este partajat doar între ei, de exemplu, o parolă sau o cheie secretă de cifrare. O altă posibilitate este de a prezenta o serie de provocări la care doar utilizatorul legitimat trebuie să poată răspunde.

O semnătură numerică este mijlocul obișnuit de identificare, deoarece asociază o parte (un utilizator sau un echipament) cu un secret comun. Alte metode de identificare simultană și de autentificare a utilizatorilor umani folosesc caracteristici fiziologice și comportamentale distincte, cum ar fi: amprentele degetare, amprentele vocale, forma retinei, forma mâinii, semnătura, mersul și așa mai departe. ♦

3.4 Identificarea biometrică

Sistemele biometrice utilizează algoritmi de recunoaștere a modelelor privind caracteristicile fiziologice și/sau comportamentale specifice ale indivizilor.

Sistemele biometrice de identificare recunosc identitatea unui individ prin potrivirea caracteristicilor extrase dintr-o imagine biometrică cu o intrare într-o bază de date a șabloanelor.

Identificarea este folosită în majoritatea cazurilor în criminalistică și aplicațiile de executare a legii.

Sistemele de verificare, din contra, autentifică identitatea unei persoane prin compararea unei imagini biometrice recent capturate cu cea a șablonului biometric al persoanei, stocat în memoria sistemului, sau în baza cartelei de identitate.

Verificarea este operațiunea care controlează accesul la resurse, cum ar fi un cont bancar sau un sistem de plăți.

3.4 Identificarea biometrică

Există două categorii principale de caracteristici biometrice.

- **modele de comportament și abilități dobândite, cum ar fi vorbirea, scrisul de mână sau tiparele de apăsare de taste;**
- **caracteristici fiziologice, cum ar fi trăsăturile faciale, morfologia irisului, textura retiniană, geometria mâinilor sau amprentele digitale.**

Metodele bazate pe mers, miros sau genetică, utilizând acid deoxiribonucleic (ADN), au aplicații limitate pentru sistemele de i-plată.

Metodele, care utilizează modele vasculare, amprente de palmă, vene de palmă sau caracteristici ale urechii, nu au fost aplicate la scară largă în i-comerț.

Acuratețea unui sistem biometric este de obicei măsurată în termeni de rată de acceptare falsă (este acceptat un intrus) și rată de respingere falsă (este refuzat accesul unui utilizator legitim). Aceste rate sunt interdependente și sunt ajustate în funcție de nivelul necesar de securitate. Alte măsuri disting între rata erorii de decizie și cea a erorii de potrivire, în cazurile în care sistemul permite încercări multiple sau include mai multe modele pentru același utilizator.

3.4 Identificarea biometrică

Alegerea unui anumit sistem de identificare biometrică depinde de mai mulți factori, cum ar fi:

1. Precizia și fiabilitatea identificării sau verificării. Rezultatul nu trebuie să fie afectat de mediu sau de îmbătrânire.

2. Costul instalării, întreținerii și funcționării.

3. Scala aplicabilității tehnicii; de exemplu, recunoașterea scrierii de mână nu este utilă pentru persoanele analfabete.

4. Ușurința de utilizare.

5. Reproductibilitatea rezultatelor; în general, caracteristicile fiziologice sunt mai reproductibile, decât cele comportamentale.

6. Rezistența la contrafaceri și atacuri.

3.4 Identificarea biometrică

Atacurile asupra sistemelor biometrice pot fi grupate în două categorii principale:

- **comune tuturor sistemelor informatice;**
- **specifice biometriei.**

Atacurile comune tuturor sistemelor informatice sunt:

1. Atacă canalele de comunicare pentru a intercepta transferurile de date, de exemplu, pentru a atenua imaginea utilizatorului a biometriei în cauză sau a caracteristicilor extrase din acea imagine. Informațiile furate pot fi reluate ca și cum ar proveni de la un utilizator legitim.

2. Atacurile diferitelor module informatice, cum ar fi manipularea fizică a senzorilor sau introducerea de programe dăunătoare (*malware*) la modulul de extragere a caracteristicilor sau la modulul de intercalare.

3. Atacă baza de date unde sunt stocate șabloanele.

3.4 Identificarea biometrică

Atacurile specifice biometriei sunt:

1. Biometriile false, cum ar fi un deget fals, o copie a unei semnături sau o mască de față, sunt prezentate senzorului.

2. Repetarea atacurilor pentru a ocoli senzorul cu semnale biometrice numerizate anterior.

3. Suprascrierea procesului de extragere a caracteristicilor cu un atac răuvoitor (*malware*) pentru a permite atacatorului să controleze caracteristicile ce urmează a fi procesate.

4. Manipularea șablonului extras din semnalul de intrare înainte de a ajunge la potrivitor (*matcher*). Acesta este cazul dacă respectivele nu sunt colocate și sunt conectate printr-o rețea la distanță mare. Atacul este mai facil dacă metoda de codificare a caracteristicilor într-un șablon este cunoscută.

5. Atac asupra potrivitorului, astfel încât să producă rezultate preselectate.

6. Manipularea șablonului stocat, în special dacă acesta este stocat pe o cartelă inteligentă care urmează să fie prezentată sistemului de autentificare.

7. Interceptarea și modificarea șablonului extras din baza de date pe drumul său către potrivitor.

8. Ignorarea deciziei finale prin modificarea rezultatelor afișării.

3.5 Autentificarea participanților

Scopul autentificării participanților este de a elimina/reduce riscul ca intrușii sub aparențe legitime să poată urmări operațiuni neautorizate.

Când participanții utilizează un algoritm de cifrare simetric, aceștia sunt singurii care partajează o cheie secretă.

Algoritmul garantează, teoretic, confidențialitatea mesajelor, identificarea corectă a corespondenților și autentificarea acestora. Serverele de distribuire a cheilor acționează și ca servere de autentificare (AS), iar buna funcționare a sistemului depinde de capacitatea tuturor participanților de a proteja cheia secretă.

Când participanții utilizează un algoritm cu chei publice, un utilizator este considerat autentic atunci când acel utilizator poate dovedi că deține cheia privată ce corespunde cheii lui publice.

Un certificat emis de o autoritate de certificare indică faptul că certifică asocierea cheii publice (și, prin urmare, cheia privată corespunzătoare) cu identitatea recunoscută. În acest mod, identificarea și autentificarea continuă în două moduri diferite, identitatea cu semnătura numerică și autentificarea cu un certificat.

3.5 Autentificarea participanților

Deși aceeași cheie publică a unui participant ar putea servi și la cifrarea mesajului adresat acelui participant (serviciu de confidențialitate) și la verificarea i-semnăturii documentelor pe care participantul le transmite (servicii de integritate și identificare), în practică, o cheie publică aparte (diferită) este utilizată pentru fiecare set de servicii.

În conformitate cu cadrul de autentificare definit de Recomandările ITU-T X.500 (2001) și X.811 (1995), autentificarea simplă poate fi realizată prin unul din următoarele mijloace:

1. **Numele și parola în text clar.**
2. **Numele, parola și un număr aleatoriu sau o ștampilă de timp (temporală), cu verificarea integrității printr-o funcție hash.**
3. **Numele, parola, un număr aleatoriu și o ștampilă de timp, cu verificarea integrității folosind o funcție hash.**

3.5 Autentificarea participanților

O autentificare puternică necesită o infrastructură de certificare care include următoarele entități:

1. Autoritățile de certificare să confirme cheile publice ale utilizatorilor cu certificate "sigilate" (adică semnate cu cheia privată a autorității de certificare) după verificarea identității fizice a proprietarului fiecărei chei publice.

2. O bază de date cu date de autentificare (directoriu) care conține toate datele referitoare la cheile private de cifrare, cum ar fi valoarea acestora, durata valabilității și identitatea proprietarilor. Orice utilizator ar trebui să poată interoga o astfel de bază de date pentru a obține cheia publică a corespondentului sau pentru a verifica valabilitatea certificatului pe care l-ar prezenta corespondentul.

3. O autoritate de numire sau înregistrare care poate fi distinctă de autoritatea de certificare. Principalul său rol este definirea și atribuirea unor nume distincte unice diferiților participanți. ♦

3.6 Controlul accesului

Controlul accesului este procesul prin care numai entitățile autorizate au acces la resursele definite în politica de control al accesului.

Este folosit pentru a contracara amenințarea cu operațiuni neautorizate, cum ar fi utilizarea neautorizată, divulgarea, modificarea, distrugerea datelor protejate sau refuzul de a oferi servicii utilizatorilor legitimi.

Recomandarea ITU-T X.812 (1995) definește cadrul pentru controlul accesului în rețele deschise.

În consecință, controlul accesului poate fi exercitat cu ajutorul unui mecanism de autentificare de sprijin la unul sau mai multe dintre următoarele straturi ale modelului OSI ISO:

- **Rețea;**
- **Transport;**
- **Aplicație.**

În funcție de strat, acreditările corespunzătoare de autentificare pot fi certificate X.509, tichete Kerberos, identitate simplă și perechi de parole.

3.7 Controlul accesului

Există două tipuri de mecanisme de control al accesului:

- **bazate pe identitate;**
- **bazate pe rol.**

Controlul accesului bazat pe identitate utilizează identitatea autentificată a unei entități pentru a determina și a impune drepturile sale de acces.

Controlul accesului bazat pe roluri folosește privilegii de acces în funcție de locație și de contextul acesteia. Astfel, în definirea politicii de acces pot fi luați în considerare factorii adiționali, de exemplu, puterea algoritmului de cifrare, tipul operației solicitate sau timpul zilei.

Controlul accesului bazat pe roluri oferă un mijloc indirect de acordare de privilegii prin trei faze distincte:

- **definirea rolurilor;**
- **atribuirea privilegiilor rolurilor;**
- **distribuirea rolurilor între utilizatori.**

Acest lucru facilitează menținerea politicilor de control al accesului deoarece este suficient de modificat definiția rolurilor pentru a permite actualizări globale fără a revizui distribuția de sus în jos.

3.7 Controlul accesului

La nivelul Rețea, controlul accesului în rețelele IP se bazează pe filtrarea pachetelor utilizând informațiile protocolului din antetul pachetului, în special adresele IP sursă și destinație și numerele de port sursă și destinație.

Controlul accesului este realizat prin "întreruperea liniei" de către un intermediar certificat sau o i-barieră care interceptează și examinează toate fluxurile de date înainte de a le permite să continue.

Intermediarul este localizat astfel între client și server.

În plus, i-bariera poate fi încărcată cu alte servicii de securitate, cum ar fi cifrarea traficului pentru confidențialitate la nivelul Rețea sau verificarea integrității utilizând semnăturile numerice.

De asemenea, poate inspecta fluxurile de date de intrare și de ieșire înainte de a le transmite pentru a aplica politicile de securitate ale unui anumit domeniu administrativ.

Cu toate acestea, intervenția părții terțe de încredere trebuie să fie transparentă pentru client.

3.7 Controlul accesului

Succesul filtrării pachetelor este vulnerabil la spionarea pachetelor dacă informațiile despre adresă nu sunt protejate și dacă pachetele individuale sunt tratate independent de celelalte pachete ale aceluiași flux.

Ca soluție, i-bariera poate include un server proxy sau o poartă (*gateway*) de nivel Aplicație care implementează un subset de funcții specifice aplicațiilor.

Serverul proxy este capabil să inspecteze toate pachetele în lumina transferurilor anterioare ale aceluiași flux înainte de a permite trecerea lor în conformitate cu politica de securitate în vigoare.

Prin filtrarea i-poștei de intrare și ieșire, a transferurilor de fișiere, a pachetelor de aplicații web și altele, porțile de aplicații pot bloca operațiile neautorizate și pot proteja împotriva codurilor rău intenționate, cum ar fi virușii. Aceasta se numește inspecție statală.

Filtrul utilizează o listă de cuvinte cheie, dimensiunea și natura atașamentelor, textul mesajului și așa mai departe.

Configurarea porții este o operațiune delicată deoarece intervenția porții nu trebuie să împiedice operarea zilnică.

3.7 Controlul accesului

A treia abordare este centralizarea gestionării controlului accesului pentru un număr mare de clienți și utilizatori cu diferite privilegii, cu un server dedicat.

Au fost definite mai multe protocoale pentru a reglementa fluxurilor între elementele de rețea și serverele de control al accesului.

RFC 6929 (2013) specifică Funcția de autentificare la distanță din serviciul utilizator (*Remote Authentication Dial in User Service* - RADIUS) pentru autentificarea clientului, autorizare și colectarea informațiilor contabile ale apelurilor.

În RFC 1492 (1993), Cisco a descris un protocol numit Terminal Access Controller Access System (TACACS), care a fost ulterior dezvoltat în TACACS+.

Atât RADIUS, cât și TACACS+ necesită o cheie secretă între fiecare element de rețea și server.

Clientul RADIUS se află în cadrul serverului de control al accesului, în timp ce serverul se regăsește într-un directoriu X.509 prin intermediul protocolului Lightweight Directory Access Protocol (LDAP).

Atât autentificarea de la server la client, cât și autentificarea de la utilizator la client sunt în afara domeniului RADIUS.

3.7 Controlul accesului

Sistemele comerciale folosesc 2 abordări de bază pentru autentificarea utilizatorilor: parola de unică folosință și răspuns-provocare.

Într-un sistem tipic cu parolă de unică folosință, fiecare utilizator are un dispozitiv care periodic generează un număr (parola de unică folosință) utilizând timpul curent, numărul de serie al cartelei și o cheie secretă păstrată în dispozitiv. Referința de timp a Serverului de control al accesului trebuie să fie sincronizată cu cartela, astfel încât serverul să poată regenera un număr identic.

În sistemele „răspuns-provocare", utilizatorul introduce un număr de identificare personal pentru a activa autentificatorul *handheld* (HHA) și apoi pentru a iniția o conexiune la un server de control al accesului. Serverul de control al accesului, la rândul său, oferă utilizatorului un număr aleatoriu (o provocare), iar utilizatorul introduce acest număr într-un dispozitiv portabil pentru a genera un răspuns unic. Acest răspuns depinde atât de provocare, cât și de unele chei secrete distribuite între dispozitivul utilizatorului și server. Acesta este returnat la serverul de control al accesului pentru a compara cu răspunsul așteptat și pentru a decide în consecință.

În comerțului mobil, deseori este folosită autentificarea cu doi factori. Situl web, ce trebuie accesat, trimite un text către utilizatorul care solicită acces cu un nou cod și o parolă de fiecare dată când utilizatorul încearcă să se autentifice.♦

3.8 Refuzuri/întreruperi în prestarea serviciului

Atacurile de refuz/întrerupere în prestarea serviciului (*denial-of-service attacks*) împiedică utilizarea normală a rețelei prin blocarea accesului utilizatorilor legitimi la resursele de rețea, prin încărcarea stațiilor cu sarcini suplimentare sau inutile pentru a împiedica reacția necesară la cererile legitime sau încetinirea timpului de răspuns mai jos de limitele satisfăcătoare.

Refuzul în prestarea serviciului rezultă din căderea controlului accesului.

Datele de control al rețelei și datele de utilizator partajează aceleași lățimi de bandă fizice și logice; iar IP este un protocol fără conexiune în care nu se aplică conceptul de control al accesului.

De aceea, la dimensiuni mari ale rețelei, traficul de control al rețelei poate să ocupe o parte semnificativă din lățimea de bandă disponibilă. Mai mult, pachetele inoportune sau prost intenționate ale utilizatorilor pot să conducă la căderea unei componente de rețea (de ex., ruter).

În cazul atacurilor distribuite orientate la refuzul în prestarea serviciului (*distributed denial-of-service* – DDD), un număr relativ mare de stații compromise pot trimite pachete inutile către o componentă-victimă în aceiași perioadă de timp, afectând astfel resursele sau lățimea de bandă sau ambele ale victimei.

3.8 Refuzuri/întreruperi în prestarea serviciului

Deoarece IP nu separă traficul utilizatorilor de cel al rețelei, cea mai bună soluție este identificarea tuturor cu certificate de încredere. Cu toate acestea, autentificarea tuturor transferurilor de date mărește sarcina de procesare, care poate fi excesivă în aplicațiile comerciale. În acest sens, mecanismele de protecție se vor dezvolta de la caz la caz.

De exemplu, epuizarea resurselor din cauza atacului SYN poate fi atenuată prin limitarea numărului de conexiuni TCP în așteptare, reducând intervalul de timp pentru sosirea pachetului ACK, înainte de a opri stabilirea conexiunii, și blocarea pachetelor către exterior care au surse din exterior.

O altă cale este reechilibrarea încărcării de procesare între cele două părți, solicitând sursei să rezolve un puzzle sub forma unor probleme cifrografice simple înainte de alocarea resurselor necesare pentru a stabili o conexiune. Pentru a evita atacurile prin repetare (replay), aceste probleme sunt formulate folosind ora curentă, un secret de server și informații suplimentare din solicitarea clientului. Această abordare necesită totuși programe pentru rezolvarea puzzle-urilor specifice fiecărei aplicații încorporate în exploratorul sursei.♦

3.9 Nerepudierea

Nerepudierea – serviciu ce împiedică o persoană, care a realizat un act, să-l refuze mai târziu, în parte sau în ansamblu.

Elementele de bază ale nerepudierii includ:

- **i-semnarea documentelor;**
- **intervenția unui terț în calitate de martor;**
- **ștampilarea timpului;**
- **numerele de ordine.**

Generarea și verificarea dovezilor necesită adesea intervenția uneia sau a mai multor entități externe părților la tranzacție, cum ar fi: un notar, un verificator și un adjudecător al litigiilor.

Recomandarea ITU-T X.813 (1996) definește un cadru general pentru nerepudierea în sistemele deschise. Serviciul cuprinde măsurile:

- **generarea probelor;**
- **înregistrarea dovezilor;**
- **verificarea dovezilor generate;**
- **recuperarea și reverificarea probelor.**

3.9 Nerepudierea

Există două tipuri de servicii de nerepudiere:

1. Nerepudierea la sursă. Acest serviciu protejează receptorul, împiedicând expeditorul să refuze trimiterea mesajului.

2. Nerepudierea la destinație. Acest serviciu joacă rolul invers al funcției precedente. Protejează expeditorul demonstrând că destinatarul a primit mesajul.

Amenințările la nerepudiere includ compromiterea cheilor sau modificarea sau distrugerea neautorizată a probelor. În cifrografia cu chei publice, fiecare utilizator este singurul proprietar al cheii private. Astfel, dacă întregul sistem nu a fost spart, un utilizator dat nu poate respinge mesajele care sunt însoțite de semnătura sa numerică.

Nerepudierea nu este ușor de realizat în sistemele care utilizează cifrografia simetrică. Un utilizator poate nega faptul că a trimis mesajul susținând că receptorul a compromis secretul partajat sau că serverul de distribuție a cheilor a fost atacat cu succes. O parte terță de încredere ar trebui să verifice fiecare tranzacție pentru a putea depune mărturie în caz de contestație.

Nerepudierea la destinație poate fi obținută folosind aceleași mecanisme, dar în sens invers.♦

3.10 Gestionarea cheilor cifrografice

Gestionarea sigură ține următoarele acțiuni privind cheile cifrografice :

- **producerea;**
- **stocarea;**
- **distribuirea;**
- **utilizarea;**
- **retragerea;**
- **ștergerea;**
- **arhivarea.**

SP 800-57 (Institutul Național de Standarde și Tehnologie, 2012c) este o recomandare din partea NIST, care oferă îndrumări privind gestionarea cheilor. Fiecare parte este adaptată la o anumită audiență:

Partea 1 este pentru dezvoltatorii de sisteme și administratorii de sistem;

Partea 2 vizează proprietarii de sisteme sau aplicații;

Partea 3 este mai generală și vizează instalatorii de sisteme, utilizatorii finali, precum și persoanele care iau decizii de cumpărare.

Generarea și păstrarea cheilor

Generarea unei chei trebuie făcută în mod aleatoriu și la intervale regulate, în funcție de gradul de securitate necesar.

Protecția cheilor stocate are un aspect fizic și un aspect logic. Protecția fizică constă în stocarea cheilor în safeuri sau în clădirile protejate cu acces controlat, în timp ce protecția logică se realizează prin cifrare.

În cazul algoritmilor de cifrare simetrică, este stocată numai cheia secretă.

Pentru algoritmii cu chei publice, se stochează:

- **cheia privată și ceia publică ale utilizatorului;**
- **certificatul utilizatorului;**
- **o copie a cheii publice a autorității de certificare.**

Certificatele și cheile pot fi stocate pe discul fix al autorității de certificare, dar există un risc de posibile atacuri sau de pierderi din cauza unei defecțiuni a echipamentelor.

În cazul cartelelor microprocesor, informațiile ce țin de securitate, cum ar fi certificatul și cheile, sunt inserate în timpul personalizării cartelei.

Accesul la aceste informații este apoi controlat cu un cod confidențial. ♦

Distribuirea cheilor

Politica de securitate definește modul în care cheile sunt distribuite entităților respective. Distribuirea manuală prin poștă sau expediție specială este o operație lentă și costisitoare, care ar trebui utilizată doar pentru distribuirea cheii rădăcină a sistemului. Aceasta este cheia, utilizată de distribuitorul de chei, pentru a trimite cheile fiecărui participant.

Un sistem automat de distribuție a cheilor trebuie să îndeplinească următoarele criterii de securitate:

- **confidențialitate;**
- **identificarea participantului;**
- **integritatea datelor – dând dovadă că cheia nu a fost modificată în timpul transmiterii sau că nu a fost înlocuită cu o cheie falsă;**
- **autentificarea participanților;**
- **nerepudierea.**

Utilizarea, retragerea și înlocuirea cheilor

Duplicarea neautorizată a unei chei legitime reprezintă o amenințare la adresa securității distribuției cheilor. Pentru a preveni acest tip de atac, un parametru unic poate fi concatenat cu cheia, cum ar fi o ștampilă de timp sau un număr de ordine care crește monotonic (până la un anumit modulo).

Riscul compromiterii unei chei crește proporțional cu timpul și cu utilizarea. De aceea, cheile trebuie înlocuite în mod regulat fără a provoca întreruperea serviciului. O soluție comună, care nu impune o încărcare semnificativă, este distribuirea cheilor de sesiune pe aceleași canale de comunicare ca și cele utilizate pentru date utilizator. De exemplu, în protocolul SSL/TLS, schimburile inițiale furnizează elementele necesare pentru a forma chei care ar fi valabile în timpul sesiunii la îndemână. Aceste elemente criptate cu o cheie secundară, numită cheie de cifrare a cheilor, pentru a păstra confidențialitatea acestora.

Serviciile de distribuție a cheilor au autoritatea de a revoca o cheie înainte de data expirării după o pierdere a cheii sau din cauza comportamentului defectuos al utilizatorului.♦

3.11 Schimbul cheilor secrete: Kerberos

Kerberos este cel mai cunoscut sistem pentru schimbul automat de chei utilizând cifrarea simetrică.

Kerberos cuprinde serviciile de identificare și autentificare online, precum și controlul accesului utilizând cifrografia simetrică. El permite accesul de gestionare la resursele unei rețele deschise de la noduri nesecurizate, cum ar fi gestionarea accesului studenților la resursele unui centru de calcul universitar (fișiere, imprimante, etc.).

Kerberos a fost opțiunea de autentificare implicită începând cu Windows 2000. Începând cu Windows Vista și Windows Server 2008, implementarea de către Microsoft a protocolului de autentificare Kerberos permite utilizarea cifrării AES 128 și AES 256 cu protocolul de autentificare Kerberos.

RFC 4120 (2005) oferă o prezentare generală și specificațiile protocolului Kerberos. Versiunea 1.13.12 a fost publicată în mai 2015.

3.12 Gestionarea certificatelor

Într-o certificare centralizată, o autoritate de certificare centrală eliberează certificatele autorităților de certificare subordonate sau intermediare, care la rândul lor certifică alte autorități secundare sau entități finale.

Aceasta este desemnată ca certificare X.509, folosind numele recomandării relevante din partea ITU-T.

Recomandarea ITU-T X.509 este identică cu ISO/IEC 9594-8, un standard comun de la ISO și IEC.

A fost aprobată inițial în noiembrie 1988, iar cea de-a șaptea ediție a fost publicată în octombrie 2012.

Aceasta este una dintre seria de specificații ITU-T și ISO/IEC comune care descriu arhitectura și operațiunile infrastructurilor de chei publice (PKI).

Unele sisteme de comunicații fără fir, cum ar fi IEEE 802.16, utilizează certificate X.509 și cifrare cu chei publice RSA pentru a efectua schimburi de chei.

3.12 Gestionarea certificatelor

Când un server primește o cerere semnată cu un algoritm de chei publice, acesta trebuie să autentifice mai întâi identitatea declarată care este asociată cheii. Apoi, va verifica dacă entitatea autentificată are permisiunea de a efectua acțiunea solicitată. Ambele verificări se bazează pe un certificat pe care o autoritate de certificare a semnat-o.

Certificarea și gestionarea certificatelor reprezintă pietrele de temelie ale i-comerțului prin rețele deschise.

Certificarea poate fi descentralizată sau centralizată.

Certificarea descentralizată utilizează PGP sau OpenPGP. Fiecare utilizator determină gradul de încredere acordat cheii publice și atribuie un nivel de încredere în certificatul pe care proprietarul cheii publice a emis-o.

Acest mod de operare elimină vulnerabilitatea la atacurile asupra unui punct central și împiedică abuzul potențial al unei singure autorități.

3.12 Gestionarea certificatelor

După ce a primit printr-o rețea deschisă o cerere cifrată utilizând cifrografia cu chei publice, un server trebuie să îndeplinească următoarele sarcini înainte de a răspunde solicitării:

1. Să citească certificatul prezentat.

2. Să verifice semnătura de către autoritatea de certificare.

3. Să extragă cheia publică a solicitantului din certificat.

4. Să verifice semnătura solicitantului în mesajul de solicitare.

5. Să verifice valabilitatea certificatului prin comparare cu Lista de revocare a certificatului (CRL).

6. Să stabilească calea de certificare între certificatul de cheie publică care trebuie validat și o autoritate recunoscută, de exemplu autoritatea de bază. Calea de certificare - sau lanțul de încredere - pornește de la o entitate finală și se termină cu autoritatea care validează calea (autoritatea de certificare rădăcină).

7. Să extragă numele solicitantului.

8. Să determine privilegiile pe care le deține solicitantul.

3.12 Gestionarea certificatelor

Certificatul permite îndeplinirea sarcinilor 1-7 din lista de mai sus. În cazul plăților, ultimul pas constă în verificarea datelor financiare referitoare la solicitant, în special dacă contul menționat are fonduri suficiente. În general, problema este mult mai complexă. Metoda cea mai directă este atribuirea unei chei fiecărui privilegiu, ceea ce sporește complexitatea gestionării cheilor.

3.12 Gestionarea certificatelor

Aplicații bancare

O bancă își poate certifica propriii clienți pentru a le permite accesul la contul lor prin Internet.

Odată ce a fost acordat accesul, operația va continua ca și cum clientul ar fi în fața unui ATM.

Interoperabilitatea certificatelor bancare se poate realiza prin acorduri interbancare, similare cu cele care au permis interoperabilitatea cartelelor bancare.

Fiecare instituție financiară își certifică clienții proprii și se asigură că celelalte instituții vor onora acest certificat.

Ca principale victime ale fraudei, instituțiile financiare au colaborat pentru a-și stabili propriile infrastructuri de certificare. ♦

3.13 Autentificarea părților

După ce a obținut calea de certificare și cheia publică autentificată a celeilalte părți, X.509 definește trei proceduri de autentificare a părților:

- **autentificare unidirecțională (cu o singură cale – *one-way*);**
- **autentificare bidirecțională (*two-way*);**
- **autentificarea tridirecțională (*three-way*).**

Autentificarea unidirecțională are loc prin transferul anumitor informații de la utilizatorul *A* la cel *B*.

Procedura de autentificare bidirecțională adaugă la schimburile unidirecționale anterioare schimburi similare, dar în sens invers.

Protocoalele pentru autentificarea tridirecțională introduc un al treilea schimb de la *A* la *B*. Avantajul este evitarea marcajului de timp și, în consecință, a unei terțe părți de încredere. ♦

3.14 Securitatea la spargeri

Factori privind securitatea la spargeri

La nivel de sistem, securitatea unui sistem depinde de mulți factori, cum ar fi

1. Rigurozitatea criteriilor de autentificare.

2. Gradul de încredere în autoritatea de bază și/sau autoritățile de certificare intermediare.

3. Puterea acreditărilor entității finale (de exemplu, pașaport, certificat de naștere, permis de conducere).

4. Rezistența algoritmilor cifrografici utilizați.

5. Forța protocoalelor principale de stabilire a cheilor.

6. Grijă cu care entitățile finale (adică utilizatorii) își protejează cheile.

Acesta este motivul pentru care proiectarea sistemelor de securitate care asigură gradul de securitate dorit necesită calificare înaltă, expertiză și atenție la detalii.

De exemplu, în cazul în care stațiile de lucru nu sunt configurate corespunzător, utilizatorilor le poate fi refuzat accesul sau mesajele de i-poștă semnate pot părea nevalabile

3.14 Securitatea la spargeri

Piețe subterane pentru parole

Răspândirea i-comerțului a fost corelată cu numărul de persoane care trebuie să se conecteze la diferite sisteme și aplicații. În multe cazuri, sistemele de autentificare sunt diferite, astfel încât utilizatorii trebuie să gestioneze un număr din ce în ce mai mare de parole, mai ales că multe sisteme obligă utilizatorii să-și înlocuiască periodic parolele. Din ce în ce mai mult, utilizatorii se confruntă cu problema creării și memorării mai multor nume de utilizatori și parole și, adesea, reutilizarea parolelor.

Una din consecințele tuturor acestor factori este creșterea criminalității informatice, alimentată de piețele subterane de parole furate și de unelte (*bots*) pentru a încerca autentificările automate în multe situri web, încercând parolele într-un fișier până la obținerea accesului. ♦

3.14 Securitatea la spargeri

Vulnerabilități de cifrare

În timp ce proprietățile teoretice ale algoritmilor cifrografici sunt importante, este esențial și cum sunt implementate și folosite fundamentele cifrografiei. Atacurile de forță brută – atacatorul încearcă în mod sistematic toate cheile de cifrare posibile până la obținerea celei care va dezvălui textul clar.

Tabelul oferă timpul estimat pentru atacurile de forță brută de succes cu căutări exhaustive pe algoritmi de cifrare simetrici cu lungimi diferite ale cheilor pentru starea la zi a tehnologiei.

Lungimea cheilor, biți	Timpul estimat pentru spargere
56–64	Câteva zile
112–128	Câțiva ani
256	Câteva zeci de ani

Vulnerabilități de cifrare

O cheie de mulți biți este condiție necesară, dar nu suficientă pentru cifrarea simetrică sigură. Cele mai frecvente tipuri de atacuri cifrologice includ:

1. Atacuri asupra textului cifrat, presupunând că textul clar are o structură cunoscută, de ex. prezența sistematică a unui antet cu un format cunoscut (mesajele de i-poștă) sau repetarea unor cuvinte cheie cunoscute.

2. Atacurile cu text selectat cifrat cu cheie necunoscută pentru a deduce cheia înseși.

3. Atacurile prin reluarea mesajelor legitime vechi pentru a evita mecanismele de apărare și pentru a scurta cifrarea.

4. Atacurile prin interceptarea mesajelor (omul-în-mijloc) între cele două părți. După interceptarea unui schimb al unei chei secrete, de exemplu, intrusul va putea descifra mesajele schimbate, în timp ce participanții cred că întrețin o comunicare în siguranță. Atacatorul poate, de asemenea, injecta mesaje false care ar fi tratate ca fiind legitime de către cele două părți.

5. Atacurile prin măsurarea duratei cifrării, a emisiilor electromagnetice ș.a. pentru a deduce complexitatea operațiunilor și, prin urmare, forma lor.

6. Atacurile asupra rețelei informatice în sine, de exemplu, spargerea DNS, pentru a direcționa traficul către un sit fals.

Vulnerabilități de cifrare

7. Menținerea parolei sau a cheilor din memoria virtuală.

8. Nici o verificare a succesivității corecte a operațiilor; aceasta este, de exemplu, cazul când depășirea de către flux a tamponului poate provoca deficiențe de securitate.

9. Utilizarea incorectă a unui protocol poate conduce la navigarea unui autentificator prin text. De exemplu, IETF RFC 2109 (1997) - acum depășită - specifică faptul că atunci când autentificatorul este stocat într-un cookie, serverul trebuie să stabilească steagul securizat în antetul "cookie" astfel încât clientul să aștepte înainte de a returna cookie-ul până când o conexiune securizată a fost stabilită cu SSL/TLS. S-a constatat că unele servere web au neglijat să creeze acest steag, negând astfel această protecție. De asemenea, autentificatorul poate fi preluat dacă i-aplicația clientului continuă s-o utilizeze chiar și după ce autentificarea a reușit.

Poate fi necesară protecția fizică a întregului sistem cifrografic (cabluri, calculatoare, cartele inteligente, etc.). De exemplu, îndoirea fibrei optice are drept rezultat dispersarea a 1% -10% din puterea semnalului; prin urmare, dispozitivele optice acustice bine plasate pot capta modelul de difracție pentru o analiză ulterioară.

Vulnerabilități de cifrare

„Atacurile de accesare a resurselor cross-app” (*cross-app resource access attacks* – XARA). Sistemele de operare, cum ar fi OS X și iOS, încearcă să izoleze aplicațiile una de cealălată, chiar și atunci când rulează același utilizator ("*sandboxing*"), pentru a împiedica un program malware sau compromis să dăuneze celorlalți. Pentru a face schimb de informații una cu alta, aplicațiile folosesc canale de comunicare interproces.

Implementarea sandboxing de către Apple are mai multe defecte care pot fi exploatate pentru a permite unei aplicații rău intenționate să obțină acces neautorizat la datele sensibile ale altor aplicații (de exemplu, parole) și să utilizeze resursele în mod secret. De exemplu, serviciul de gestionare a creditelor de la Apple, denumit „lanțchei” (*keychain*), permite fiecărei aplicații să înregistreze acreditările utilizatorului (parolele utilizatorului, cheile secrete și certificatele), dar nu oferă o modalitate ușoară de a determina proprietarul unui element de lanțchei și/sau pentru a autentifica respectivul proprietar înainte de a acorda accesul la elementul respectiv.

Profitând de acest defect, o aplicație rău intenționată poate aloca în mod secret atributele cheie pentru aplicațiile care nu au fost încă instalate, cu speranța că utilizatorul le va instala mai târziu. În acest caz, aplicația rău intenționată va avea acces deplin la acreditive (informațiile de acreditare - *credencials*). Dacă aplicația țintă este deja instalată, aplicația rău-intenționată poate șterge elementul corespunzător de lanțchei și îl poate înlocui cu o nouă listă de control al accesului sub controlul atacatorului. Deci, când aplicația țintă actualizează acreditările utilizatorului, acestea vor fi divulgate atacatorului.

Vulnerabilități de cifrare

Este, de asemenea, posibil de profitat de alte detalii de implementare care nu țin direct de cifrare. De exemplu, atunci când un program șterge un fișier, majoritatea sistemelor de operare comerciale elimină doar intrarea corespunzătoare din fișierul index. Acest lucru permite recuperarea fișierului, cel puțin parțial, cu i-programe de tip "*off-the-shelf*". Singurul mijloc prin care se garantează eliminarea totală a datelor este de a rescrie sistematic fiecare dintre biții pe care fișierul eliminat îi folosea. În mod similar, utilizarea memoriei virtuale în sistemele comerciale expune o altă vulnerabilitate, deoarece documentul secret poate fi momentan în clar pe disc.

Sistemele de i-comerț destinate publicului larg trebuie să fie ușor accesibile și la prețuri accesibile. În consecință, se vor face numeroase compromisuri pentru a îmbunătăți timpul de răspuns și ușurința utilizării. Cu toate acestea, dacă se pornește de la principiul că, mai devreme sau mai târziu, orice sistem este susceptibil de atacuri neașteptate cu consecințe neprevăzute, este important ca sistemul să permită detectarea atacurilor și acumularea dovezilor acceptate de personalul de aplicare a legii și de instanțele judecătorești. Punctul principal este de a avea o definiție precisă a tipului de amenințări preconizate și posibilele atacuri. O astfel de evaluare realistă a amenințărilor și riscurilor permite o înțelegere precisă a ceea ce ar trebui protejat, împotriva cui și pentru cât timp. ♦

Phishing, Spoofing și Pharming

Phishing-ul (pescuitul), spoofing-ul și pharming-ul sunt folosite pentru a păcăli utilizatorii și a le dezvălui în mod voluntar acreditările (*credentials*). Acești termeni sunt neologisme care descriu diverse trucuri ale utilizatorilor. Ele sunt adesea folosite interschimbabil.

Phishing este un mesaj fals pentru a înșela utilizatorii să dezvăluie entităților neautorizate acreditările lor (numere de cont bancar, parole, detalii cartele de plăți, etc.). Termenul a fost inventat pentru a indica faptul că infractorii "pescuiesc" detaliile bancare online ale clienților prin i-poștă. Expeditorul poate să se implice în companii de renume, cum ar fi intermediari financiari sau bănci, și să exploateze întreaga gamă de emoții și credulități umane, de la obligația față de prietenii aflați în primejdie în străinătate, de teama unor mandate de arestare și până la lăcomie și ignoranță.

De exemplu, mesajele presupuse de o bancă ar avertiza destinatarul că există o problemă de securitate cu contul destinatarului care necesită o atenție imediată și apoi solicită destinatarului să facă clic pe o legătură, care pare legitimă, sub orice pretenție falsă (pentru a restabili un cont compromis, pentru a verifica identitatea, etc.).

Phishing, Spoofing și Pharming

Prin utilizarea formularelor HTML în corpul textului sau al unui manipulator de evenimente JavaScript, referința văzută în mesajul de i-poștă este diferită de destinația actuală a referinței. După ce se face clic pe referință, se deschide o fereastră care conține situl real sau o copie falsă a sitului web al băncii și se vor cere detalii despre cont sau datele de autentificare. Atunci când utilizatorul introduce datele de identificare, acestea sunt captate și utilizate pentru a accesa contul utilizatorului.

După ce a convins destinatarul că mesajul de i-poștă este credibil și provenit de la o instituție de încredere, phishing-ul exploatează întreaga gamă de emoții și credulitate umane pentru a convinge destinatarul să divulge informații personale și financiare sub forma de verificare a contului, prietenul rănit într-o țară străină și așa mai departe.

În mod alternativ, un i-program rău intenționat (*malware*) poate fi injectat în dispozitivul țintă pentru a înregistra toate intrările de la tastatură și trimite periodic datele colectate la o adresă anume (*drop mail*). O formă extremă a acestui i-program este numit i-program de răscumpărare (*ransomware*), unde intrușii profită de controlul sistemului țintă, dacă nu se plătește o răscumpărare.

Phishing, Spoofing și Pharming

În februarie 2015, firma Kaspersky Lab a emis un raport în care descria modul în care grupul Carbanak a reușit să pătrundă în rețelele bancare folosind e-mailuri de tip phishing pentru a fura peste 500 de milioane de dolari de la bănci din mai multe țări și de la clienții lor privați. În unele cazuri, bancomatele au fost instruite să-și distribuie conținutul la asociați. În alte cazuri, au fost modificate bazele de date pentru a mări soldurile frauduloase pe conturile existente și apoi au preluat diferența fără cunoștința proprietarului contului.

Farming-ul constă în exploatarea sistematică a unei vulnerabilități a serverelor DNS cauzată, de exemplu, de o eroare de codare sau un i-program rău intenționat implantat. Ca rezultat, în loc de a traduce numele de Web în adresa lor IP corespunzătoare, sunt utilizate adrese IP false asociate cu un sit Web fals. Efectele spoofing-ului și ale farming-ului sunt aceleași, dar în cazul spoofing-ului victima participă, inconștient, ca și cum de bună voie, pe când în cazul farming-ului victima nu știe de redirecționare.

Cauza principală a tuturor acestor probleme este lipsa autentificării în schimburile de i-poștă. iPoșta a fost inițial destinată comunicării între colaboratorii ce se cunosc, și nu pentru utilizatorii distribuiți în peste 70 de milioane de domenii, pentru a comunica fără o examinare prealabilă. Numai soluții parțiale au fost oferite până acum.♦

3.15 PGP și OpenPGP

Pretty Good Privacy este considerat a fi sistemul comercial a cărui securitate este cea mai apropiată de cea militară. Este descris în IETF RFC 1991 (1996).

PGP cuprinde șase funcții:

1. Schimbul cheilor publice folosind RSA cu hash-ul MD5.

2. Comprimarea datelor cu ZIP, care reduce dimensiunea fișierelor și redundanțele înainte de cifrare (Reducerea dimensiunii fișierelor crește viteza atât de procesare, cât și detransmisie; totodată, reducerea redundanței face mai dificilă cifranaliza).

3. Cifrarea mesajelor cu IDEA.

4. Cifrarea cheii secrete a utilizatorului, utilizând rezumatul unei fraze în loc de parolă.

5. „Armura" ASCII pentru a proteja mesajul binar pentru orice mutilare care ar putea fi cauzată de sistemele de mesagerie pe Internet (Acest „armor" este construit prin împărțirea biților a trei octeți consecutivi în patru grupuri a câte 6 biți fiecare și apoi prin codarea fiecărui grup folosind un caracter pe 7 biți conform unui tabel dat. Se adaugă apoi o sumă de control pentru a detecta eventualele erori).

6. Segmentarea mesajelor.

3.15 PGP, OpenPGP și GnuPG

IETF nu a adoptat PGP ca standard deoarece încorporează protocoale proprietare.

OpenPGP se bazează pe PGP, dar evită aceste probleme de proprietate intelectuală.

Este specificat în RFC 4880 (2007).

RFC 6637 (2012) descrie modul de utilizare a cifrografiei curbelor eliptice (ECC) cu OpenPGP.

OpenPGP este în prezent cel mai utilizat standard de cifrare a e-mail-urilor.

Companiile și organizațiile care implementează OpenPGP au format Alianța OpenPGP pentru a o promova și pentru a asigura interoperabilitatea.

Fundația Free Software și-a dezvoltat propriul program conform OpenPGP, numit GNU Privacy Guard (GnuPG abreviat sau GPG).

GnuPG este disponibil gratuit, împreună cu întregul cod sursă, sub licența GNU General Public License (GPL).