UNIVERSITATEA DE MEDICINĂ, FARMACIE, ȘTIINȚE ȘI TEHNOLOGIE “GEORGE EMIL PALADE” DIN TÂRGU-MUREȘ
FACULTATEA DE INGINERIE ȘI TEHNOLOGIA INFORMAȚIEI
SPECIALIZAREA: INFORMATICĂ

Platformă de certificare bazată pe o infrastructura de chei publice

Conducător științific:

Lect. univ. dr. ing. Bogdan Crainicu

Absolvent:

Suciu Oana Mădălina

TÂRGU MUREȘ
2021

UNIVERSITATEA DE MEDICINĂ, FARMACIE, ȘTIINȚE ȘI TEHNOLOGIE “GEORGE EMIL PALADE” DIN TÂRGU-MUREȘ
FACULTATEA DE INGINERIE ȘI TEHNOLOGIA INFORMAȚIEI
SPECIALIZAREA: INFORMATICĂ

LUCRARE DE LICENȚĂ

Conducător științific:
Lect. univ. dr. ing. Bogdan Crainicu

Absolvent:
Suciu Oana Mădălina

TÂRGU MUREȘ
2021

CUPRINS

CAPITOLUL 1

INTRODUCERE

În ziua de azi majoritatea informațiilor sunt răspîndite prin intermediul internetului, de aceea trebuie acordată o atenție sporită securităţii datelor.

Criptografia este ştiinţa care protejează informaţiile faţă de terţi neautorizaţi. Însă, pe lîngă criptarea efectivă a datelor, un rol extrem de important îl constituie mecanismele de autentificare, integritatea datelor, validitatea sursei datelor și non-repudierea.

Lucrarea tratează platformele bazate pe chei publice (PKI-Public Key Infrastructure) şi propune o soluţie completă de autoritate de certificare:

– certificate digitale de client;

- funcţii de semnare, reînnoire și revocare a certificatelor;

- depozitarea certificatelor digitale pe un token criptografic;

- utilizarea de certificate în scop de autentificare şi de semnare digitală;

Aplicaţia permite semnarea de documente de orice tip, cu condiţia ca semnatarul documentelor să fie client al autorităţii de certificare.

CAPITOLUL 2

TEHNICI SI ALGORITMI DE CRIPTARE

Criptologia reprezintă subcîmpul matematicii care se ocupă de modificarea sau conversia informațiilor, pentru a proteja sau restricționa accesul la anumite informații sensibile. Aceasta cuprinde domeniul criptografiei, care reprezintă criptarea datelor, dar şi domeniul criptanalizei care reprezintă „o artă” a descifrării informaţiilor criptate.

Termenul de criptografie are origini greceşti şi este compus din cuvintele “kryptós”, care are sensul de ascuns şi “gráphein” reprezentînd scrierea [1] .

Funcțiile oferite de criptografie sunt:[2]

- **Integritatea datelor**: garantează nemodificarea sau nealterarea pe parcursul comunicării a informaţiei transmise, acest lucru realizîndu-se prin ataşarea la informaţia distribuită a unui mesaj de control de lungime fixa care poate fi recreat de către destinatar şi permiţîndu-i acestuia să verifice posibilele modificări a informaţiei primate.
- **Autentificarea:** posibilitatea receptorului de a verifica sursa exactă a informaţiei primite, prevenind astfel furtul de identitate
- **Confidenţialitatea:** oferă imposibilitatea unei entități neautorizate de a avea acces la informaţia reală.
- **Non-repudierea:** se folosește pentru a “obliga” semnatarul să îşi recunoască informația şi pe viitor, adică să nu poată nega sau denigra acea informație..

Din punct de vedere tehnic, criptografia ascunde informaţiile utile de terţe priviri neautorizate. Dacă se cunosc elementele necesare (cum ar fi: cheia criptografică şi mecanismul de criptare), atunci putem decripta informaţia securizată, astfel încît să avem acces la aceasta.

Pentru criptarea informaţiilor este nevoie de o cheie de criptare care reprezintă o secvenţă de caractere cu care se criptează sau se decriptează informaţia dorită.

Cele două mari tipuri de criptografie pe care le întîlnim sunt:

- Criptografia simetrică

- Criptografia asimetrică

Criptografia simetrică reprezintă acel tip de criptografie care foloseşte o cheie unică, atît pentru operaţiunile de criptare cît şi pentru cele de decriptare. Condiţia, foarte importantă, că acest tip de criptografie să funcţioneze corect este că toţi partenerii de comunicaţie să deţină cheia simetrică, în prealabil, iar distribuirea ei să se facă printr-o metodă securizată.[3]

Criptografia asimetrică reprezintă acel tip de criptografie care foloseşte o pereche de chei, legate matematic între ele, şi anume o Cheie Privată şi o Cheie Publică. Aceste două chei se generează într-o anumită ordine: Se generează cheia privată iar din cheia privată se generează foarte uşor Cheia Publică. Inversul operaţiei, adică generarea Cheii Private din Cheia Publică este foarte dificil de realizat, de aceea nici nu se consideră, practic, posibilă, de la un anumit nivel. Din punct de vedere al rolurilor acestor chei, Cheia Privată rămîne întotdeauna la proprietar, iar Cheia Publică se distribuie la partenerii de comunicaţie.[4]

Iniţializarea criptării simetrice, în mod securizat, cu ajutorul criptografiei asimetrice se realizează în felul următor (exemplu simplificat - unele protocoale au elemente suplimentare de securitate):

- Presupunem 2 parteneri de comunicaţie: Alice şi Bob.
- Alice vrea să comunice securizat cu Bob, dar amîndoi se afla la o distanţă de 5000 Km unul de altul.
- Pentru a putea comunica securizat, amîndoi trebuie să deţină aceeaşi cheie simetrică.
- În acest sens, Alice cere Cheia Publică a lui Bob (care poate fi distribuită de Bob), iar Alice alege o cheie secretă, pe care o criptează cu Cheia Publică a lui Bob, şi i-o trimite lui Bob, astfel încît doar Bob (cu cheia lui Privată) poate ca să Decripteze secretul trimis de Alice.
- Astfel, în acest moment, ambii parteneri deţin secretul comun (Alice generîndu-l iar Bob primindu-l securizat, cu ajutorul cheii asimetrice), iar de acum încolo poate începe comunicaţia securizată folosind acelaşi secret comun.

Încă de la începutul dezvoltării scrisului, oamenii au încercat să ascundă aceste scrisuri pentru a nu le face valabile pentru oricine. Acest lucru este dovedit de către inscripţiile pe

piatră, papirus, tăbliţe, care arată că egiptenii, evreii şi asirienii au conceput sisteme criptografice pentru ascunderea adevăratelor informaţii.

Prima utilizare a criptografiei înregistrate a fost făcută de către spartani, care foloseau un dispozitiv special de cifrare (scena) pentru comunicarea secretă între comandanţii armatei militare. Totuşi, oamenii care au înţeles funcţionalităţile criptografiei au fost arabii, descoperind criptanaliza.

2.1 Criptografie simetrică

Criptografia simetrică reprezintă cea mai simplă modalitate de criptare și implică folosirea unei chei secrete comune. Aceasta mai reprezintă și singura formă de criptografie cunoscută pînă în anul 1976, fiind prima formă de criptografie modernă.

Astfel, prin criptografie simetrică se înțelege acea modalitatea prin care doi utilizatori care doresc să comunice securizat folosesc ca modalitate de criptare aceeași cheie. Această metodă presupune ca cei doi utilizatori să dețină în prealabil cheia respectivă. (Fig. 1)

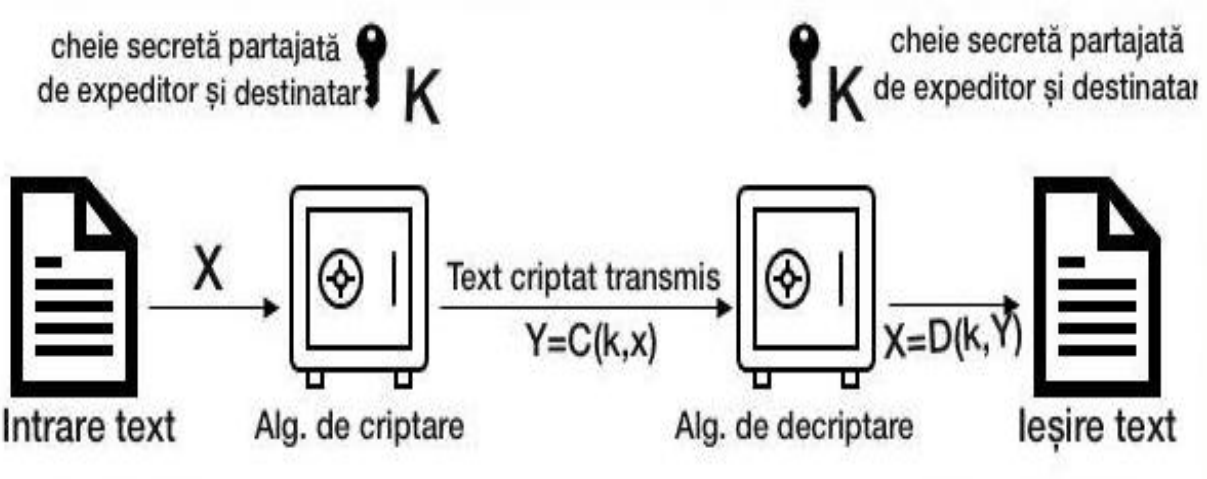

Fig. 1 – Criptare simetrică [1]

Cel mai relevant exemplu, care datează cu aproximație din anul 50 î.Hr, este cifrul lui Caesar, fiind una dintre cele mai simple și cunoscute tehnici de criptare. Mai exact, reprezintă un cifru în care fiecare literă din textul inițial este înlocuită cu o altă literă care este poziționată în alfabet la o distanță fixă față de cea înlocuită. Spre exemplu, alfabetul clasic român este “ABCDEFGHIJKLMNOPQRSTUVWXYZ”, iar, aplicînd deplasamentul de 4 obținem “EFGHIJKLMNOPQRSTUVWXYZABC” (Fig. 4). Astfel, pentru a transmite

cuvîntul “COPAC“ folosind schema sus-prezentată acesta devine “GSTEG“. În acest caz cheia este “deplasament +4“ [5] .

Datorită faptului că, pentru a folosi metoda simetrică, este necesar ca utilizatorul care primește mesajul criptat să dețină, in prealabil, cheia de criptare / decriptare, astfel, transmiterea cheii simetrice prin diverse canale devine o problemă. Acest impediment a facilitat crearea criptografiei asimetrice.(Fig.2)

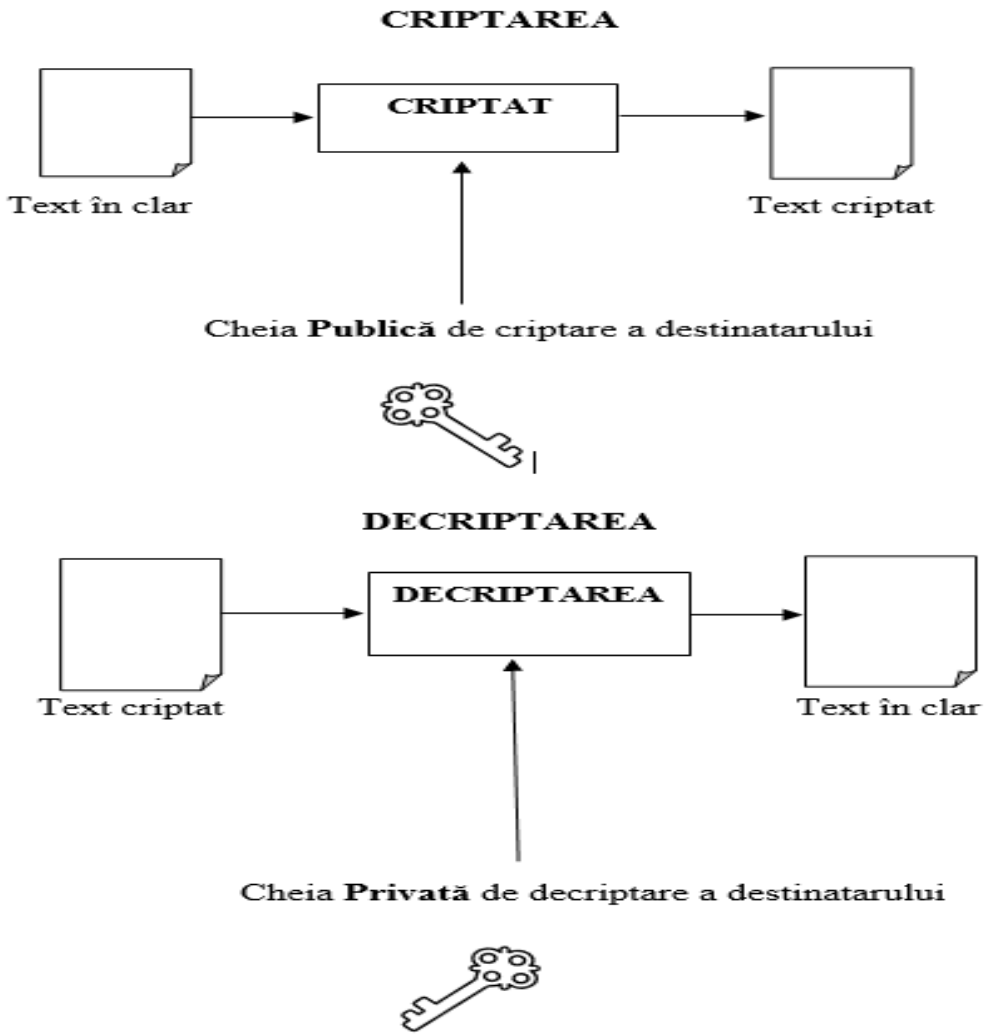

Fig.2-Text criptat prin criptare simetrica

O caracteristică a algoritmilor simetrici este reprezentată de viteza de cifrare foarte mare, comparativ cu algoritmii asimetrici, cît, faptul că cifrarea unei cantități mari de informații este ușor acceptată.

Criptografia simetrică, datorită tehnologizării cît și al necesității de a comunica într-un mediu cît mai sigur face ca aceasta să fie folosită preponderent în medii interne în care utilizatorii se cunosc și sunt de încredere. Problematica pe care acest tip de criptografie nu a

reușit să o rezolve este aceea de a facilita transmiterea cheii printr-un canal securizat, destinatarul mesajului fiind în imposibilitatea de a-l deschide fără cheie. Astfel, securitatea algoritmului simetric depinde de dimensiunea cheii și posibilitatea ca această cheie să rămînă secretă, disponibilă doar destinatarului.[6]

În urma unui experiment s-a ajuns la concluzia că folosind o mașină complexă, în paralel, este posibil ca în 60 de ore să descifreze o cheie de 56 de biți în care a fost criptat un text clar. Datorită acestuia, la începutul anilor 2000, NIST (National Institute of Standards and Technology) a selectat algoritmul AES (Rijndael) să fie noul standard în criptografia simetrică.

AES (Advanced Encryption Standard) conține trei blocuri de cifruri: AES-128, AES-192, AES-256, toate fiind adoptate dintr-o colecție mai mare publicata de catre Rijndael, autorul cifrului. Fiecare cifru AES conține un bloc de 128 biți, cu mărimea cheilor de 128, 192 si 256.[7]

Algoritmul AES este bazat pe principul de rețea de permutare al substituției, fiind eficient si la nivel hardware dar si software, necesitînd putină memorie.

O secvență AES se executa astfel:

• Expansiune de Chei folosind Programarea de Chei Rijndael

• Runda inițială:

1. Se adaugă cheia rundei

• Runde:

1. SubBytes – o substituție non-lineară in care fiecare bit este substituit cu altul din tabel (Fig. 3)

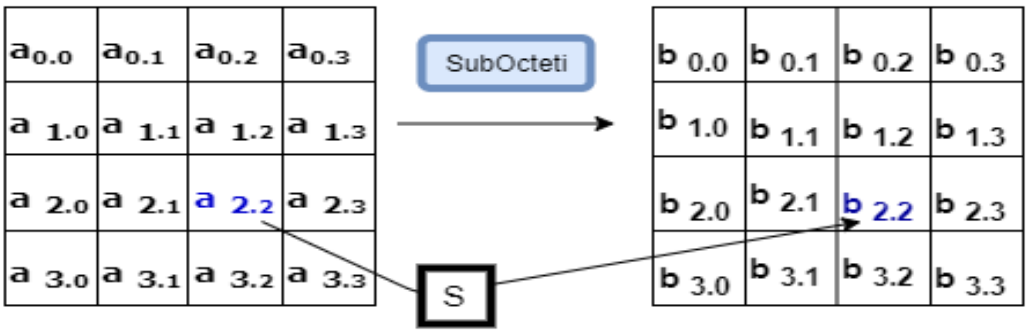

Fig. 3-SubBytes : AES [8]

2. ShiftRows – un pas transpozițional unde fiecare rînd al stării este mutat

ciclic un număr stabilit de pași (Fig. 4)

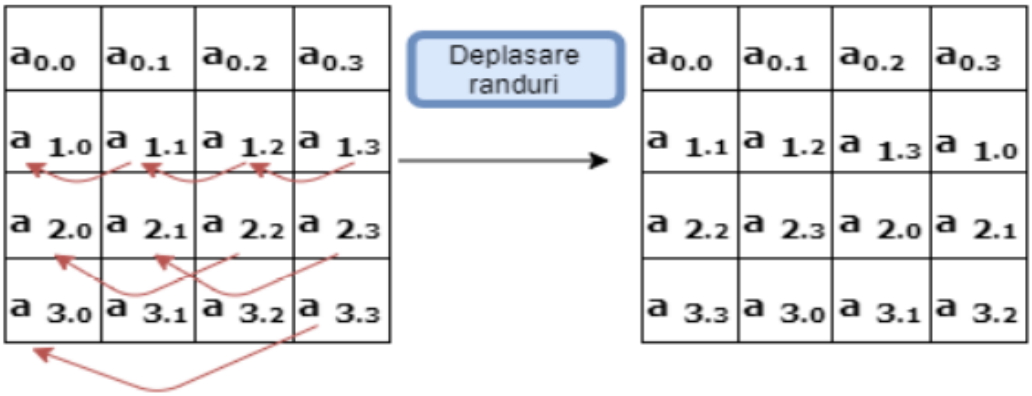

Fig. 4- ShiftRows : AES [8]

3. MixColumns – o operație de mixare care operează pe coloanele de stare, combinînd cei patru octeți din fiecare coloană.(Fig. 5)

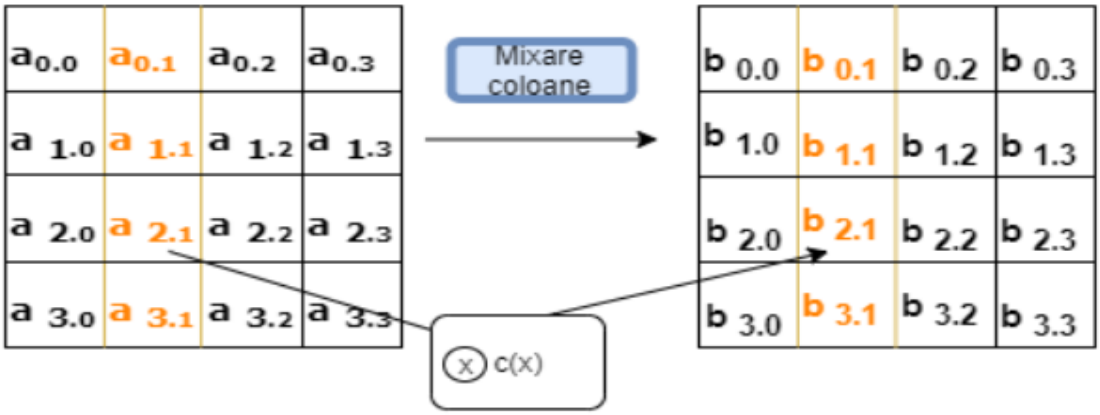

Fig.5-MixColumns:AES [8]

4. AddRoundKey – fiecare bit din stare este combinat cu cheia rundei; fiecare cheie a rundei este derivată din cheia cifrului folosind un programator de chei.(Fig.6)

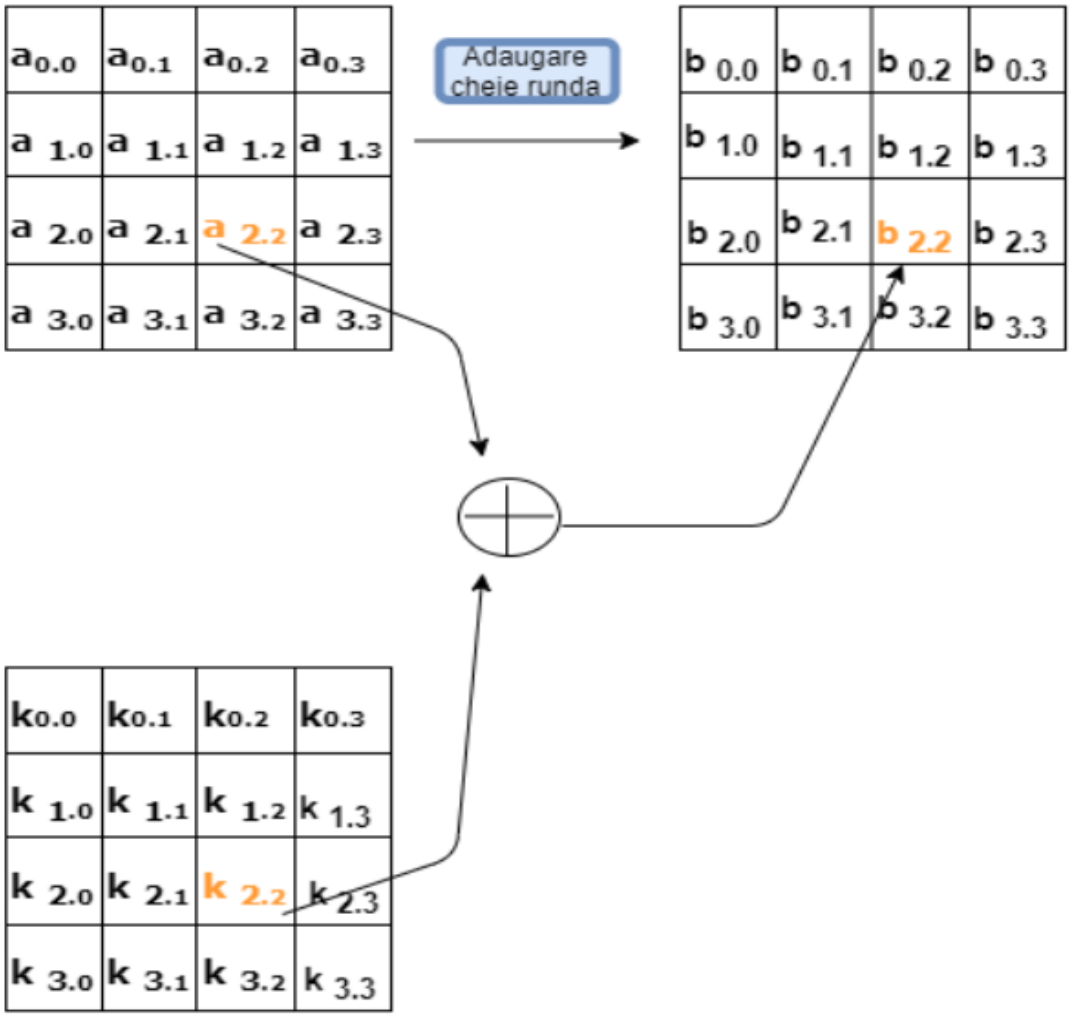

Fig.6-AddRoundKey [8]

2.2 Criptografie asimetrica

Criptografia asimetrică reprezintă acel tip de criptografie care utilizează perechi de chei, chei publice și private. Cel care deține o astfel de pereche de chei, utilizatorul, publică cheia publică astfel încît oricine o poate folosi pentru a-i transmite un mesaj acestuia. Matematic, aceste două chei sunt corelate, dar, cheia privată nu poate fi obținută din cheia publică (Fig. 7) [9].

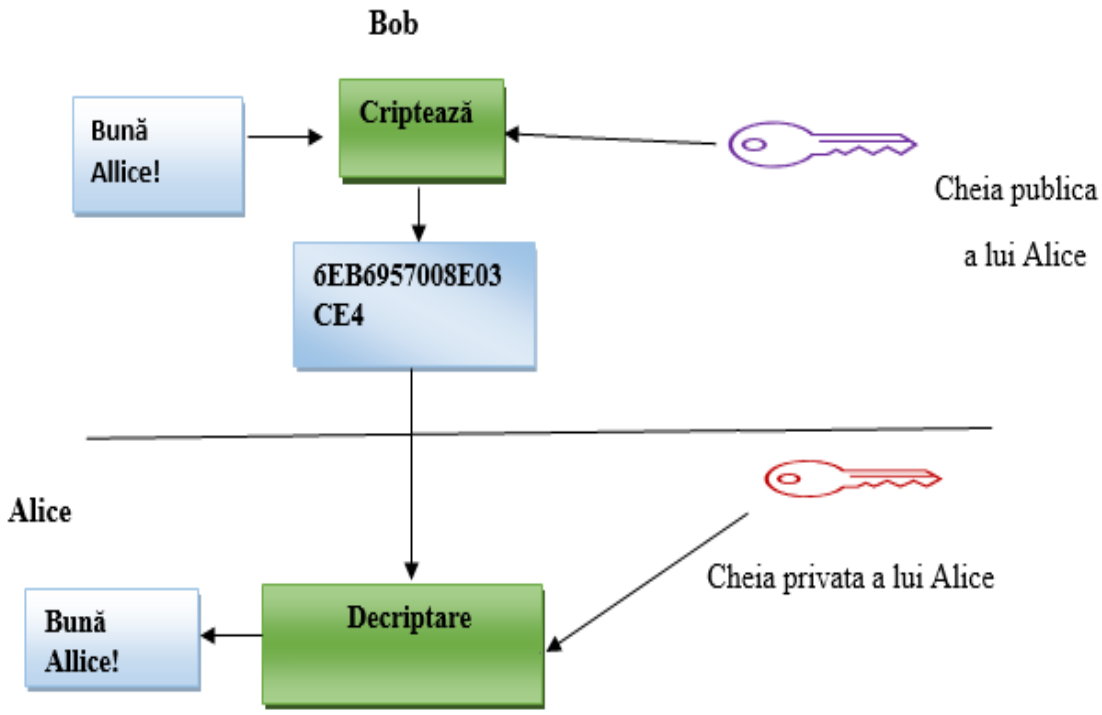

Fig.7-Schema operațiilor in criptografia asimetrică

Acest proces, simplificat, poate fi comparat cu o cutie poștală, spre exemplu, oricine poate introduce plicuri în aceasta, dar, doar cel care deține cheia cutiei poate avea acces la cutie.

Acest tip de criptografie mai este numit și criptografie cu chei publice. Cheile sunt legate matematic una de cealaltă facilitînd ca din cheia privată să fie dedusă cheia publică dar niciodată invers. Tehnic, este posibil ca din cheia publică să fie obținută cheia publică dar doar după zeci sau chiar sute de ani de încercări folosind cele mai puternice calculatoare.

Criptografia asimetrică reprezintă şi un picior de plecare în comunicaţiile criptate, deoarece rezolva foarte uşor problema predistribuirii cheii simetrice. În mod normal, este mult mai fezabila comunicaţia cu chei simetrice (deoarece reduce stresul computaţional asupra criptării/decriptării), însă, dacă partenerii de comunicaţie se afla la distanţa iar cheia simetrică nu poate fi distribuită securizat, atunci se apelează la criptografia asimetrică pentru a iniţializa criptarea simetrică (ex: SSH, SFTP, HTTPS, etc.)

Spre deosebire de criptografia simetrică, criptografia asimetrică folosește o pereche de chei pentru operațiile criptografice, cheie privată și cheie publică. Generarea acestor chei are loc dintr-un număr aleator foarte mare, în urma folosirii unor funcții matematice.

Cheia publică poate fi distribuită de către deținătorul cheii private terțelor persoane doresc să interacționeze din punct de vedere criptografic. De asemenea, cheia privată este imperios necesar să rămînă în posesia proprietarului pentru a nu permite compromiterea acesteia de către alte persoane.

O altă metodă comună o reprezintă crearea unei chei secrete între două persoane, astfel, fiecare persoană generează o cheie publică și una privată, ulterior, cheia publică se transmite unul altuia. Cheia publică a uneia dintre persoane se combină cu cheia privată a celeilalte, rezultînd astfel o cheie comună.

Criptografia asimetrică elimină cu succes problema predistribuirii cheii, întîlnită la criptografia simetrică, deoarece informația de interes poate fi decriptată doar de către deținătorul cheii private.

Cele două mari ramuri ale criptografiei asimetrice sunt reprezentate de criptografia bazată pe chei publice și semnarea digitală prin intermediul cheii private a deținătorului.

Mai există posibilitatea creării unei chei private comune între doi utilizatori, astfel, fiecare persoană generează o pereche de chei, privată și publică, după care, se distribuie cheia publică unul celuilalt. După validarea cheii de către ambele părți, cheia publică a celuilalt utilizator se combină cu cheia privată, rezultînd astfel o cheie comună. Aceasta poate fi folosită ca o cheie criptografică simetrică [10].

CAPITOL 3

INFRASTRUCTURI DE CHEI PUBLICE – PKI

3.1 Principiile criptosistemelor bazate pe chei publice

Acest concept a evoluat pornind de la existența a două mari probleme ale criptării convenționale, anume, cea a distribuției cheilor, problemă care pune utilizatorii în situația de a apela la un distribuitor de chei, iar, cea de-a doua îl reprezintă imposibilitatea folosirii criptării simetrice la nivel global.

Însuși Whitfield Diffie, fondatorul algoritmului cu criptare cu cheie publică, prezintă faptul că datorită problemei expuse mai sus, anume, imposibilitatea de a deține cheia publică fără ca utilizatorul să apeleze la un distribuitor de chei neagă în esență criptografia, care are la bază abilitatea de a ține comunicarea secretă. O a doua problemă expusă de Diffie a fost aceea a semnăturii digitale ca și semnătura clasică folosită pe documentele electronice, cu aplicabilitate mai mult în domeniul militar sau comercial.

Astfel, datorită necesității creării unui nou protocol criptografic care să elimine problemele sus-menționate, Diffie și Hellman au dezvoltat o nouă metodă anume, algoritmul Diffie-Hellman, prin intermediul căruia este posibilă schimbarea de chei criptografice prin intermediul unui canal public, fiind unul dintre primele protocoale cu cheie publică concepute de Ralph Merkle și numite după Whitfield Diffie și Martin Hellman. Schimbul de chei Diffie-Hellman este alcătuit din: înțelegerea de chei Diffe-Hellman, negocierea de chei Diffie-Hellman, stabilirea de chei Diffie-Hellman, schimbul de chei și protocolul Diffie-Hellman. Schema a fost publicată în anul 1976 [11]. (Fig. 8)

Avantajele folosirii sistemului PKI sunt reprezentate de administrarea semnăturilor digitale, a documentelor, securizarea poștei, securizarea aplicaților care rulează în rețelele private sau publice, autentificare generală. Formele mai noi de criptografie se bazează pe platforma PKI.

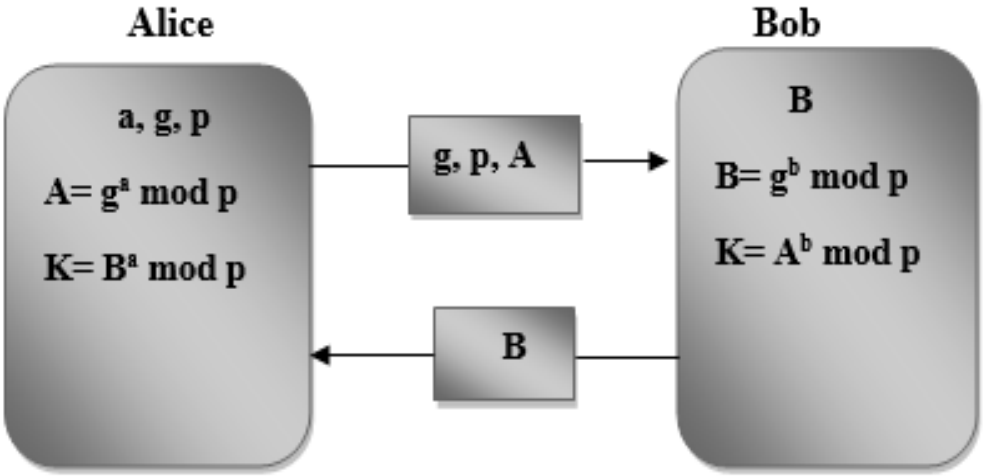

Fig.8- Protocolul Diffie-Hellmman implementat matematic

3.2 Algoritmul RSA

În anul 1977, Ron Rivest, Adi Shamir și Leonard Adleman au dezvoltat și publicat la MIT algoritmul RSA, nume compus din inițialele acestora. MIT-ului i-a fost garantat patentul U.S Patent 4,405,829 pentru "sistem și metodă de comunicare criptografică", folosind algoritmul în anul 1983. Deși Patentul a expirat în anul 2003, algoritmul a fost lansat public în anul 2000 de către RSA Security. Acest algoritm este în momentul de față cel mai cunoscut, stabil și utilizat algoritm criptografic cu chei publice, utilizat atît pentru criptare cît și pentru semnătură electronică.[12]

Algoritmul RSA reprezintă un sistem criptografic care folosește chei publice reversibile în care atît spațiul mesajelor M cît și spațiul criptogramelor C reprezintă mulțimea Zn = unde n = pq este produsul a două numere prime distincte, aleatoriu alese.

Algoritmul de generare a cheilor [13].

- Se aleg două numere prime distincte,p și q.

- Se calculează n= pq și ϕ = (p- 1)(q -1).

- Se alege aleator un număr întreg e, 1< e< ϕ astfel încît cmmdc (e, ϕ) = 1

- Prin algoritmul extins al lui Euclid se determină d 1< d< ϕ astfel încît ed = 1(mod ϕ)

- Cheia publică a lui A este (n, e), iar cheia privată este d

- n ca valoare întreagă poartă denumirea de modul, iar e și d sunt exponenții de criptare și decriptare

Algoritm criptare:

```
Input: (n,e) //cheia publica de autentificare a lui A
       m  //mesaj sub forma unei valori intregi
                Output: c // criptograma
Inceput:    c=m*e mod n;
sendA(c);
Sfarsit.
```

Algoritm decriptare:

```
                Input:  d //cheia privata
                        c  //mesaj criptat
                Output: m //mesaj decriptat
                Inceput: m = c*d mod n
                         retun m;
                Sfarsit.
```

Datorită faptului că operația este destul de costisitoare din punct de vedere alt resurselor cît și al timpului alocat, anume exponențierea modulo n, viteza algoritmului RSA este mai mică decît cea a algoritmilor care folosesc cheie secretă. Din această cauză, în sistemele de comunicație în timp real(aplicațiile de streaming video sau audio securizate), se foloseşte un sistem de criptare hibrid, mai exact algoritmul RSA se foloseşte la începutul comunicației, pentru a transmite cheia secretă de comunicație, ulterior folosită într-un algoritm cu cheie secretă, cum ar fi 3DES sau AES.

Scheme de tip "umplutură" RSA:

În practică algoritmul RSA este combinat cu diverse scheme de tip "umplutură" cu scopul de a preveni atacuri care ar putea funcționa pe RSA fără "umplutură".

- În momentul în care se criptează cu exponenți de criptare joasă cît și valori mici ale lui "m", rezultatul "m" la "e" este strict mai mic decît modulului lui "n". Astfel, textele-cifru pot fi cu ușurință decriptate folosind a "e"-a rădăcină a textului-cifru peste întregi.

- în cazul în care același mesaj sub formă necriptată este trimis la "e" sau mai mulți recipienți, după criptare, iar, destinatarii împart același exponent "e" dar p", "q" și "n" diferă, prin intermediul teoremei "Chinese remainder theorem" este ușor ca mesajul să fie decriptat. Dacă atacatorul cunoaște o relație lineară între acestea, acest atac este posibil chiar dacă mesajele clare nu sunt egale.

- criptosistemul este considerat securizat în cazul în care atacatorul nu poate distinge două criptări una de cealaltă chiar dacă acesta cunoaşte clar textele. Astfel, algoritmul RSA neavînd componentă aleatoare, un atacator poate lansa un atac împotriva criptosistemului, prin criptarea textelor clare, prin folosirea cheii publice şi prin intermediul testării dacă acestea sunt egale cu cifrul-text.

Astfel, prin intermediul celor prezentate mai sus un atac cu cifru-text ales este posibil.

Implementările algoritmului RSA includ scheme de randomizare de tip “umpluturi“ în valoarea lui “m“ înainte de a-l decripta, tocmai pentru a preveni aceste probleme. Prin intermediul acestor implementări “m“ nu cade în segmentul de texte nesecurizate, astfel, un mesaj “umplut“ va fi criptat folosind un număr mare de texte clare.

Algoritmul RSA poate fi folosit şi pentru a semna un mesaj, spre exemplu Bob doreşte să-i trimită un mesaj lui Alice, acesta folosind cheia publică a lui Alice pentru a expedia mesajul. În mesajul respectiv Bob pretinde a fi Bob, dar, Alice nu are posibilitatea să verifice acest lucru deoarece la cheia ei publică au acces mai mulţi utilizatori. Astfel, folosirea algoritmului RSA pentru a verifica originalitatea mesajului este posibilă.

Folosind exemplul de mai sus, se presupune că Bob doreşte să-i trimită un mesaj criptat lui Alice. Pentru a face acest lucru, Bob poate să folosească cheia lui privată, astfel, el produce o sumă de control a mesajului, o ridică la puterea lui "dmod n" şi ataşează semnătura digitală mesajului. În momentul în care Alice primeşte mesajul, foloseşte acelaşi algoritm în conjucţie cu cheia publică a lui Bob. La fel ca şi la criptarea mesajului, Alice ridică semnătura la puterea lui "emod n" şi compară rezultatul valorii hash cu hashul actual al mesajului. Dacă valoriile primite de hash în urma aplicării formulelor corespund, însemnă că expeditorul mesajului şi cel care a plicat semnătura este Bob.

3.3 Algoritmul Diffie-Hellman

În anul 1976 Diffie şi Hellman au dezvoltat o metodă care să ducă la rezolvarea problemelor pe care criptografia clasică le avea, anume, utilizatorii sunt nevoiţi să apeleze la un distribuitor de chei, iar, cea de-a doua îl reprezintă imposibilitatea folosirii criptării simetrice la nivel global.

Astfel, s-a creat un protocol criptografic care a luat numele celor doi, anume Diffie-Hellman. Acesta face posibil ca doi utilizatori total străini unul de celălalt să aibă posibilitatea să iniţieze o conexiune securizată printr-o cheie secretă comună într-un canal plic neasigurat.

Aceasta folosește pentru decriptare un cifru de chei simetrice și se bazează pe conceptul perechii de chei publică-privată. Protocolul prin care funcționează algoritmul presupune ca fiecare parte să genereze independent cîte o cheie privată, ulterior, fiecare utilizator calculează cîte o cheie publică, aceasta fiind o funcție matematică a cheilor private respective. Pasul următor presupune schimbul de chei publice, iar, în final fiecare dintre cei doi utilizatori calculează o funcție a propriei chei private și a cheii publice deținute de celălalt utilizator. Matematic, se va ajunge la aceeași valoare, derivată din cheile private ale utilizatorilor, folosind valoare ca și cheie a mesajului.

Astfel, dacă Bob dorește să-i trimită un mesaj lui Alice, acesta îi poate transmite cheia publică fără griji, care, va fi folosită de către Alice pentru criptarea mesajului, ulterior la primirea datelor criptate Bob va folosi cheia sa privată pentru a vizualiza informațiile în clar. În cazul în care fișierul criptat cu cheia lui Bob publică ajunge la o altă persoană, în principiu deducerea cheii private din cea publică este imposibilă, chiar dacă există o legătură matematică între acestea. Cheile reprezintă șiruri de numere create după unele reguli și sunt măsurate în biți.(Fig. 9)

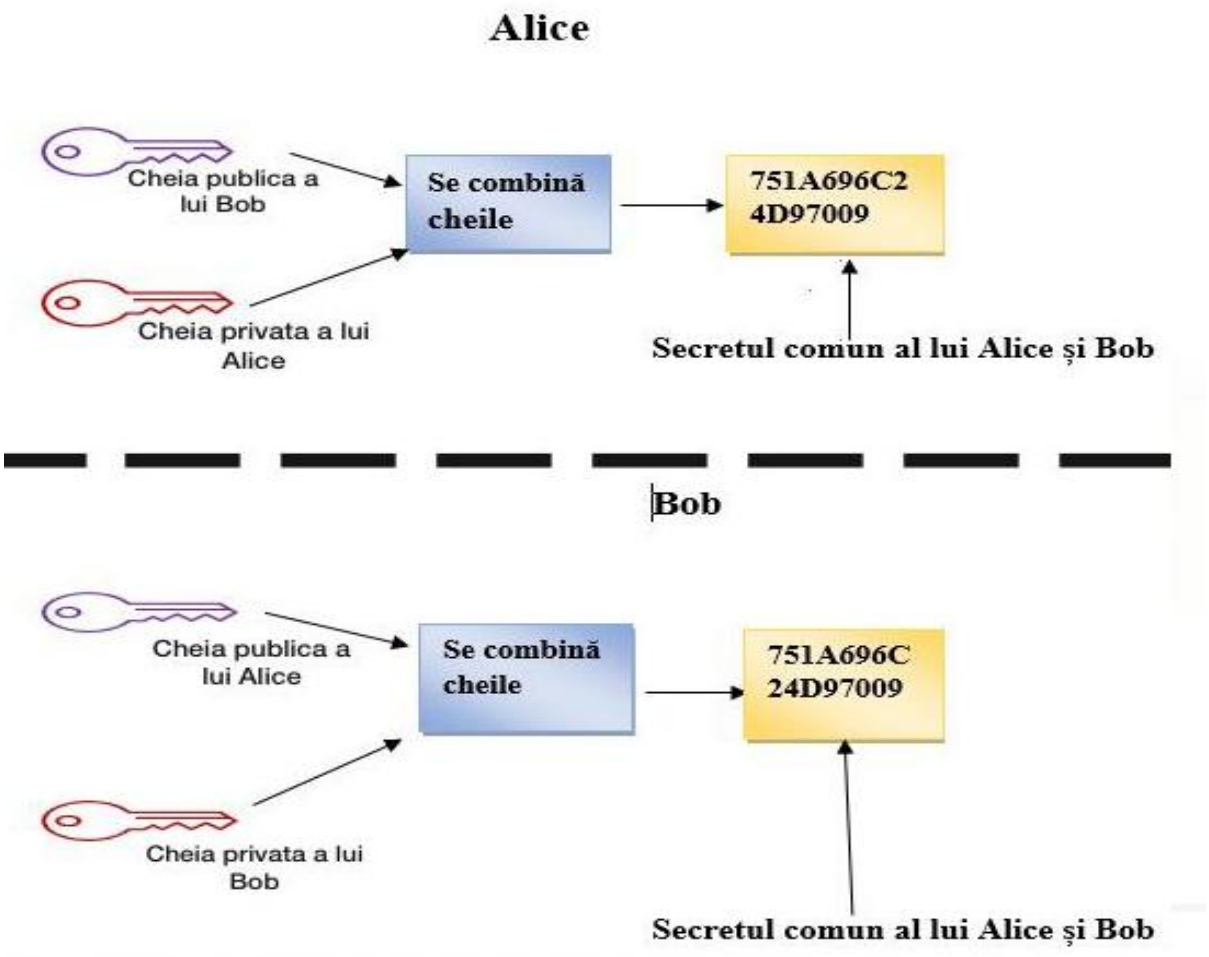

Fig.9- Crearea unei chei hibride prin algoritmul Diffie Hellman

În următorii ani Serviciul Secret Britanic a creat un nou protocol separat de către Malcom J Williamson dar a fost clasificat.

În anul 2002 i-a fost recunoscută participarea lui Ralph Merkle la inventarea criptografiei publice, iar Hellman a propus ca algoritmul să se numească Diffle-Hellman-Merkle key exchange. Acest protocol îi acreditează pe sus-numiții ca și inventatori.

Datorită faptului că în descrierea originală a algoritmului acesta nu prevede autentificarea utilizatorilor la comunicare, algoritmul este vulnerabil la atacurile de tip "om-în-mijloc". Mai exact, utilizatorul care se află în situația "om-în-mijloc" ar avea posibilitatea să schimbe două chei distincte, una cu primul utilizator, iar, cealaltă cu al doilea, prefăcîndu-se că este primul utilizator iar mai apoi al doilea, permițînd atacatorului să decripteze iar ulterior să recripteze mesajele între utilizatorii de bună credință. Datorită acestei slăbiciuni este necesară introducerea unei metode de autentificare a utilizatorilor la comunicarea securizată.

3.4 Criptografia bazată pe curbe eliptice

Neal Kobiltz și Victor S. Miller, în anul 1985, au sugerat utilizarea curbelor eliptice în criptografie, acestea, algoritmii de criptografie a curbei eliptice intrînd pe scară largă între anii 2004-2005.

Principalul beneficiu pe care criptografia cu curbă eliptică îl aduce este reprezentat de o dimensiune mai mică a cheii, reducînd astfel cerințele de stocare și transmisie, mai exact un grup de curbe eliptice ar putea oferi același nivel de securitate oferit de un sistem bazat pe algoritmul RSA cu un modul de dimensiuni mai mari și cheie mai mare, de exemplu o cheie publică cu curbă eliptică de 256 de biți ar trebui să ofere o securitate comparabilă cu o cheie publică RSA de 3072 biți. Încrederea în criptografie bazată pe curbe eliptice a crescut în 1999 și 2001, moment în care au fost standardizate pentru domeniul bancar sau semnături digitale. Pentru a înțelege importanța criptografiei bazată pe curbe eliptice, aceasta este utilizată de Agenția Națională de Securitate a Statelor United ale Americii (NSA) pentru protejarea informațiilor clasificate, inclusiv top secret cu chei de 384 biți.

Prin criptografie cu curbă eliptică (ECC) se înțelege o abordare a criptografiei cu cheie publică, care are la bază o structură algebrică a curbelor eliptice peste cîmpuri finite,

astfel, ECC facilitează folosirea unei chei mai mici față de criptografia bazată pe cîmpuri Galois simple.

Comparație între criptografia bazată pe curbe eliptice, criptografia simetrică și asimetrică: (Tabel 1)

Chei criptografia simetrică	Chei RSA	Chei ECC
80	1024	160
112	2048	224
128	3072	256
192	7680	384
256	15360	521

Tabel 1-Tabel cu recomandările NIST de dimensiuni ale cheilor[14]

3.5 Funcții criptografice de tip hash. Aplicabilitate

Funcția de tip hash reprezintă procedura prin care se preia un număr la întîmplare de date și ulterior returnează un rînd de caractere de mărime fixă, oricare ar fi mărimea mesajului, numită sumă de control. Suma de control este elaborată ca fiind unică pentru fiecare mesaj în parte, în așa fel încît orice modificare suferită de mesajul inițial să schimbe valoarea sumei de control.(Fig. 10)

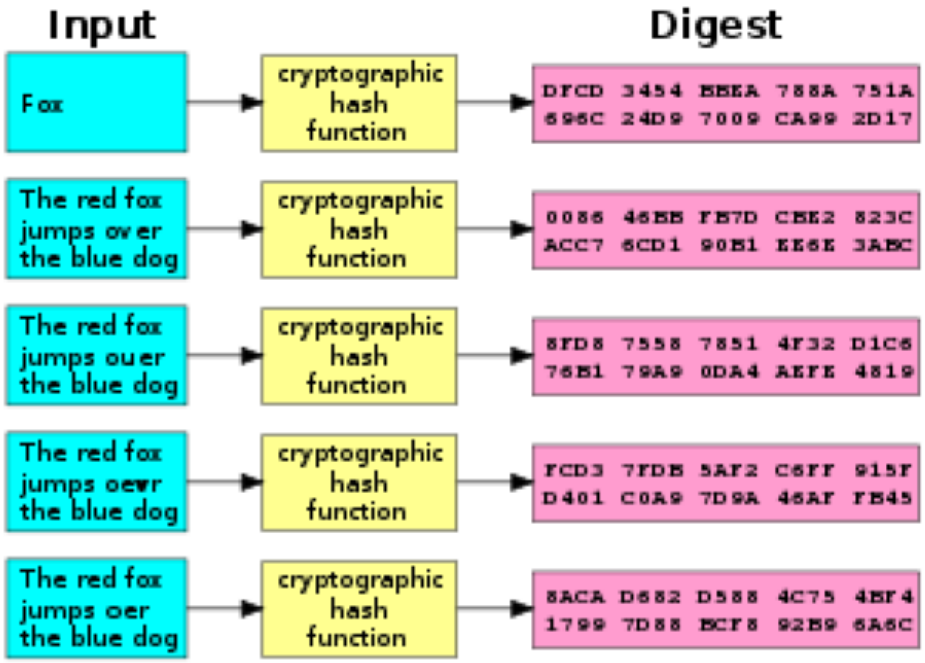

Fig. 10-Funcție criptografică hash (SHA1) [15]

Ca și proprietăți, funcția de tip hash:

- nu permite ca după modificarea mesajului inițial suma de control să fie aceeași;
- ușurința prin care este calculată suma de control pentru mesajele date;
- extragerea mesajului din suma de control nu este posibilă;
- găsirea a două mesaje diferite care să dețină aceeași sumă de control este imposibilă.

Funcțiile de tip hash sunt utilizate pentru verificarea integrității unui anumit tip de mesaj cît, pentru aplicații specifice securității informatice cum ar fi semnăturile digitale și alte forme de autentificare[16].

Principalele funcții de tip hash întîlnite în practică sunt cele într-un singur sens, cît, cele rezistente la coliziuni. Ele sunt folosite în mai multe domenii, cum ar fi facilitarea căutării într-o bază de date mare sau tabel, ca și sume de control cît și în criptografie[17].

3.6 Semnături digitale

Semnătura digitală sau electronică reprezintă un simbol digital cu aceeași valoare legală ca și semnătura clasică, fizică, folosită pentru validitatea unor documente digitale. Aceasta, în cazul în care este bine implementată, poate fi mult mai greu de falsificat decît semnătura clasică, fizică, cu condiția ca, cheia privată cu se folosește să rămînă secretă. Acestea sunt cel mai des utilizate în certificate, e-mailuri sau contracte care folosesc protocoale criptografice [18].

Față de semnăturile clasice, semnătura electronică asigură faptul că semnatarul nu poate să nege apartenența acesteia, acesta fiind responsabil din punct de vedere legal pentru documentul semnat digital. Astfel, prin intermediul cheii publice se poate verifica cu ușurință apartenența cheii private cît și autenticitatea ei.(Fig. 11)

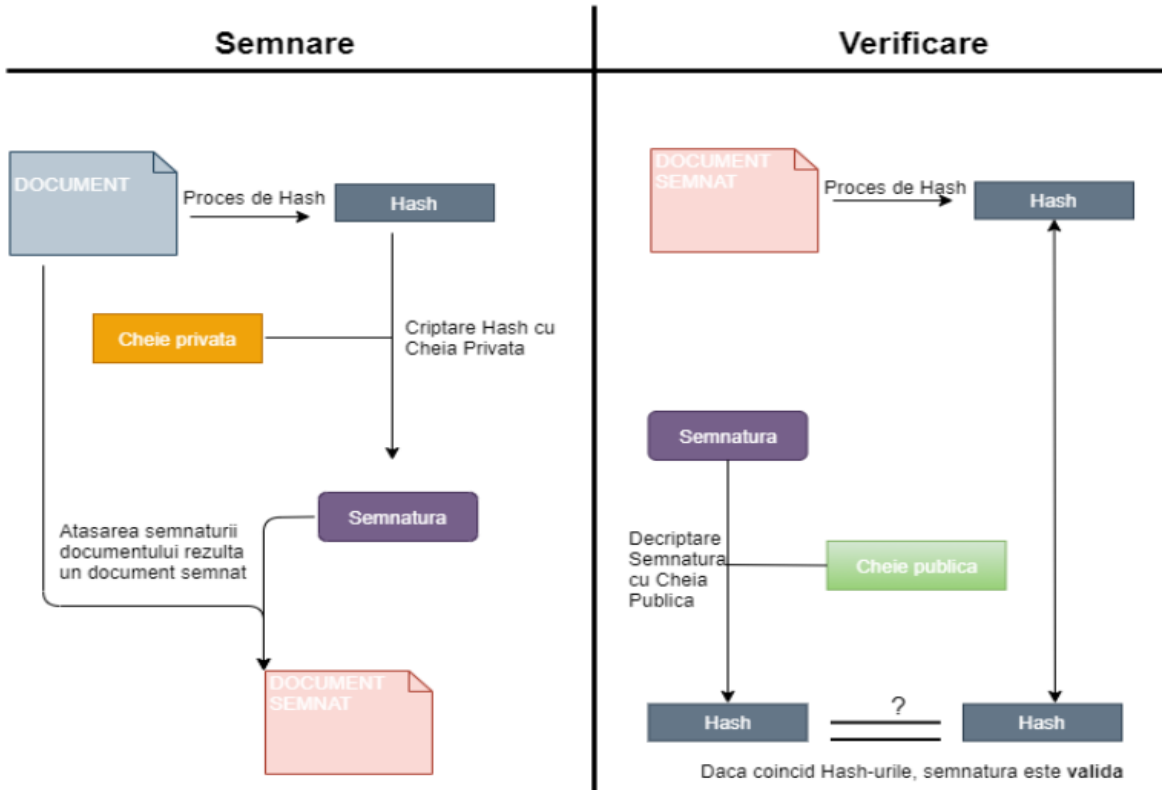

Fig.11 – Schema semnare si verificare semnatura

Pentru ca pe un document să fie aplicată o semnătura electronică este nevoie ca pe acesta să se aplice un hash, rezultatul fiind un hash unic pentru documentul respectiv. Ulterior, documentul este criptat folosind cheia privată deținută de către semnatar rezultînd semnătura digitală. Astfel, se atașează semnătura la document și rezultă un document semnat.

Pentru a verifica autenticitatea semnăturii documentului, se ia documentul semnat și se aplică un hash pe acesta, se reține hashul documentului, după care se decriptează prin intermediul cheii publice a semnatarului. Astfel, se returnează semnătura decriptată. Dacă hashul obținut prin intermediul decriptării este identic cu cel obținut din hashuirea documentului, atunci, semnătura este validă.

Semnătura digitală conține trei algoritmi:

- Algoritmul care generează cheile private, ulterior predă cheia private și cheia publică.

- Algoritmul de semnare, care produce semnătura în urma atașării mesajului și a cheii private.

- Algoritmul de verificare a semnăturii, care confirmă sau infirmă în urma atașării mesajului, cheii publice și semnăturii dacă aceasta este autentică.

Este imposibilă aflarea cheii private din cheia publică, cît, semnătura generată în urma unui mesaj fix și a unei chei private fixe trebuie să verifice cheia publică corespunzătoare.(Fig.12)

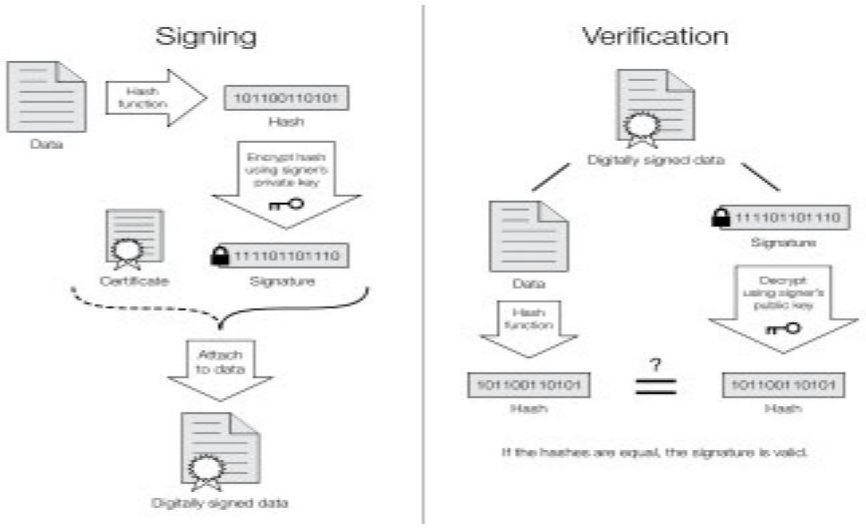

Fig.12-Semnare și verificare cu chei RSA [19]

Tehnic vorbind, semnăturile bazate pe RSA se calculează astfel, se generează chei RSA care conțin modulul “N”, produsul a două numere prime, cu întregii “e” i “d” astfel încît “ed = 1 mod (EulerPhi)(N)”.

Practic, semnarea nu se aplică pe întreg documentul ci doar pe hashul acestuia din motive de compatibilitate, funcția hash facilitînd convertirea mesajului într-un input fix, eficiență, semnătura este scurtă, economisind astfel timp datorită faptului că generarea sumei de control este mai rapidă decît semnarea cît și integritatea, anume, dacă nu se folosește funcția hash, este nevoie ca documentul să fie divizat în bucăți mai mici pentru a permite accesarea acestuia.

Semnătura digitală facilitează autentificarea celui care o folosește deoarece, din momentul în care acesta semnează cu cheia privată și distribuie cheia publică, orice persoană care deține cheia publică poate să verifice faptul că cel care deține cheia privată a semnat într-adevăr documentul respectiv. Aceasta asigură transmiterea în cel mai mic detaliu al mesajului, deoarece, semnătura digitală identifică orice modificare a unui bit, astfel, semnătura se invalidează datorită modificării sumei de control.

Principalul element de securitate al semnăturii private este reprezentat de cheia privată, care asigură integritatea semnăturii digitale.

3.6.1 Schema Elgamal

În 1984 Taher Elgamal a anunțat o nouă schemă bazată pe chei publice apropiată ca și principiu de funcționare de algoritmul Diffie-Hellman (ELGA84, ELGA85). Criptarea Elgamal este formată din algoritmul de criptare, algoritmul de decriptare și generatorul de chei. Acesta implică folosirea cheii private pentru generarea semnăturii digitale și cheia publică pentru verificarea semnăturii digitale.

Elementele principale ale semnăturii digitale Elgamal sunt reprezentate de numărul prim q și a care sunt rădăcini primitive ale lui q. Utilizatorul A generează o cheie publică/privată astfel:

```
Input:  q , a
 Output:  XA cheia privată a lui A
        {q, a, YA}.cheia publică a lui A {q, a, YA}.
  Inceput: XA=generareNumarAleator();
           YA=AXa mod q.
      return YA, XA
   Sfarsit.
```

Pentru a semna un document M, utilizatorul A prima dată calculează hasul m=H(M), astfel încît m reprezintă un întreg cuprins între 0<=m<=q-1. Ulterior A formează o semnătură

```
Input:  M //mesaj
Output:  (S1,S2) //semnatura
Inceput: m=H(M); //calculare hash
  K= generareNumarK(1<=K; K<=q-1; gcd(K, q-1)==1);
  S1=aK mod q;
  S2=K-1(m-XAS1)mod(q-1).
  return (S1,S2).
 Sfarsit.
```

De exemplu, se dă următorul cîmp prim GF(19), q=19. Cîmpul are rădăcini primitive {2,3,10,13,14,15}. Se alege a=10.

Alice generează o cheie astfel:

-Alice alege X_A=16.

-După care $Y_A=a^{X}{}_{a} \bmod q=a^{16} \bmod 19=4$.

-Cheia privată a lui Alice este 16; Cheia publică a lui Alice este {q, a, Y_a}={19, 10, 4}.

Dacă Alice dorește să semneze un document cu hash de valoare m=14.

- Alice alege K=5, număr care este relativ prim cu q-1=18.

- $S_1=a^K \bmod q=10^5 \bmod 19=3$.

- $K^{-1} \bmod(q-1)=5^{-1} \bmod 18=11$.

$-S_2=K^{-1}(m-X_AS_1) \bmod (q-1)=11(14-(16)(3)) \bmod 18= -374 \bmod 18=4$.

Bob poate să verifice semnătura astfel.

$-V_1=a^m \bmod q=10^{14} \bmod 19=16$.

$-V_2=(Y_A)^{S1}(S1)^{S2} \bmod q=(4^3)(3^4) \bmod 19=5184 \bmod 19=16$.

Astfel, semnătura este validă datorită faptului că $V_1=V_2$.

Algoritm verificare semnatura:

```
Input:  q, a,  (S1,S2)
Output:  Adevarat sau Fals daca
         semnatura este valida
Inceput: V1=ammod q.
         V2=(YA)S1(S1)S2mod q
         If(V1== V2) {
            return true
          }
         else return false
Sfarsit.
```

3.6.2 Schema Schnorr

Schema Schnorr reprezintă o semnătură digitală produsă de către algoritmul Schnorr, descris de către Claus Schnorr [SCHN 89, SCHN 91]. Această schemă minimalizează cantitatea de calcul necesar pentru a genera o semnătură, fiind cunoscută pentru eficiența și simplitatea sa. Activitatea principală necesară pentru generarea semnăturii nu depinde de mesaj și poate fi făcută în timpul în care procesorul este inactiv.

Schema are la bază folosirea unui modul prim p, cu p-1 care deține un factor q de mărimi apropiate. Astfel, p este un număr de 1024-bit și q este un număr de 160-bit, care este de asemenea lungimea valorii SHA-1 hash.

Pașii necesari pentru această schemă sunt reprezentați de generarea cheii publice sau private, astfel:

- Se alege un număr prim q și p, q fiind un număr prim al lui p-1.

- Se alege un întreg a, ca și a^q=mod p. Valoarea lui a, p și q cuprinde o cheie publică globală comună unui grup de utilizatori.

- Se alege un număr aleatoriu întreg s cu 0 < s < q. Aceasta reprezintă cheia privată a utilizatorului.

- Se calculează $v=a^{-s}$ mod p. Aceasta reprezintă cheia publică a utilizatorului.

Utilizatorul care deține atît o cheie publică cît și o cheie privată generează o semnătură digitală astfel.

- Se alege un număr aleatoriu întreg r cu 0 < r < q și se calculează $x=a^{'}$ mod p. Calculul acesta reprezintă un pas de preprocesare al mesajului M care urmează să fie semnat.

- Se concatenează mesajul cu x și hash pentru a calcula valoarea lui e, astfel:

$$e=H(M||x)$$

- Se calculează y=(r+se) mod q. Semnătura constă în perechea (e, y).

Orice alt utilizator poate verifica semnătura astfel:

- Se calculează $x^{'}=a^{y}v^{e}$ mod p.

- Se verifică: $e=(H||x^{'})$

Pentru a verifica dacă funcționează, se remarcă

Prin urmare, H (M||x')= H (M||x). [2]

Exemplu:

Alice și Bob sunt prieteni și Alice și-a anunțat toți prietenii despre faptul că deține o cheie publică și că poate primi informații sau mesaje prin intermediul acesteia. Bob crede că Alice minte, iar, aceasta dorește să demonstreze faptul că nu minte fără a-și face publică cheia privată, astfel, prin intermediul Schemei lui Schnnor Alice demonstrează:

- Se consideră următorii parametrii: p, q, a, s, v, r, x, y

unde: - p reprezintă orice număr prim

- q este un factor al lui p-1

- a astfel încît a^q= 1 mod p

Aceste variabile sunt publice astfel încît orice utilizator care dorește să le vadă are această posibilitate.

- Se dau următoarele chei

    - s reprezintă cheia privată, astfel încît 0 < s < q

    - v reprezintă cheia publică, astfel încît a^-s mod q.

Cheia publică "v" este publică împreună cu p, q și a. Cu toate acestea Alice deține informația cheii private "s".

Alice dorește să trimită un mesaj criptat "M", astfel:

- Se alege un număr aleatoriu r astfel încît 0 < r < q.

- Se calculează valoarea lui X astfel încît X=a^r mod p.

- După calcularea valorii lui X, aceasta se concatenează cu mesajul original, astfel, se concatenează M și X pentru a obține M | | X și va stoca hashul valorii în e.

    e=H(M| |X) unde H() reprezintă funcția hash.

- Se obține valoarea y astfel, y=(r+s*e) mod q.

- Se trimit următoarele către Bob:

    - mesajul M

    - semnăturile

- Bob deține următoarele informații:

    - Cheia publică a lui Alice "v".

    - Numărul prim pe care Alice l-a ales, "p"

    - "q" reprezintă factorul lui p-1, ales de Alice.

    - "a" astfel încît a^q=1 mod p, ales de Alice.

- Bob trebuie să calculeze X' astfel încît:

    X'= a^y v^e mod p

- Se cunoaște faptul că v=a^-s, astfel obținem:

    X'= a^y * a^se= a^(y-s*e)

- Se cunoaște faptul că:

    y= r + s*e, care însemnă:

    r= y - s*e

- Se înlocuiește această valoare în ecuația următoare:

Avem: X'= a^r

X=a^r

Astfel, X=X'

Cu toate acestea Bob nu cunoaşte valoarea lui "X", deoarece el a primit doar mesajul M și cheia publică “v”, p, q, și , astfel:

Se rezolvă e: e= H (M | | X')

Anterior s-a rezolvat H(M||X)), astfel, dacă cele două valori ale lui e sunt egale, înseamnă că X=X'.

3.6.3 Schema bazată pe curbe eliptice

Criptografia pe curbe eliptice, prescurtată ECC (Elliptic Curve Cryptography) a fost descoperită între anii 1986-1987 de către Neal Koblitz şi Victor Miller. La începutul anilor 90 lumea s-a arătat reticentă față de acest nou tip de criptografie, fiind lansate mai multe speculații despre practicalitatea și securitatea ECC. Ulterior, după unele cercetări s-a dovedit faptul că criptografia pe curbe eliptice este la fel de sigură precum algoritmul RSA sau schemele bazate pe DLP. Astfel, între anii 1990-2000, schema bazată pe curbe eliptice a început să fie folosită în domeniul bancar, schimburi de chei care aveau la bază curbele eliptice, semnături digitale, standarde comerciale cum ar fi TLS, IPsec.

Spre deosebire de algoritmul RSA, implementările folosite pentru ECC sunt mult mai mici și rapide, RSA sau DLP folosind chei pe 1024-3072 biți iar ECC chei pe 160-250 biți, oferind aproximativ acelaşi nivel de securitate.

3.6.4 Schema RSA-DSS

Semnătura digitală reprezintă o modalitate de autentificare a informațiilor provenite de la un utilizator de încredere. Digital Signature Standard sau prescurtat DSS reprezintă un standard federal de analizare a informaților care definește algoritmii utilizați pentru generarea semnăturilor digitale prin intermediul SHA(Secure Hash Algorithm) cu rolul de a autentifica documentele digitale[.1]

DSS oferă doar funcția de semnătură electronică fără a se axa pe criptare sau schimb de chei publice/private.(Fig. 13)

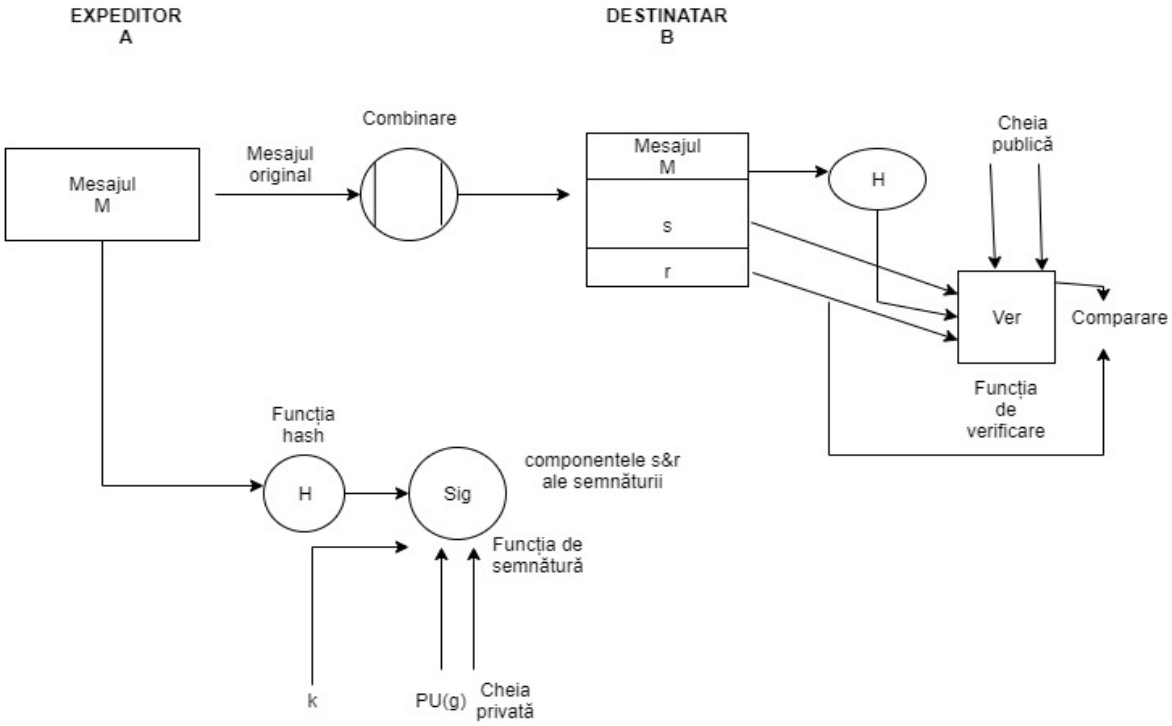

Fig.13- Verificare semnătură [20]

Pentru ca funcția reprezentată de DSS să fie îndeplinită, este nevoie ca atît utilizatorul care expediază semnătura digitală cît și receptorul să respecte cumulat următoarele:

Expeditorul:

Codul hash este generat în urma mesajului și intrările sunt date funcției de semnătură:

- Codul hash.

- Numărul aleatoriu "k" generat pentru semnătură.

- Cheia privată a expeditorului.

- O cheie publică globală, la care să aibă acces toți utilizatorii.

Prin intermediul acestor funcții se va putea determina semnătura de ieșire care conține două componente, "s" și "r". Astfel, mesajul concatenat cu semnătura este trimis receptorului.

Receptorul:

Codul hash trimis de către expeditor este generat și vericiat prin intermediul funcțiilor care preiau următoarele intrări:

- Codul hash generat.

- Semnăturile "s" și "r".

- Cheia publică a expeditorului.

- Cheia publică disponibilă pentru toți utilizatorii.

Prin urmare, dacă după compararea rezultatului funcției de verificare a semnăturii rezultă faptul că aceasta este egală cu valoarea pe care semnătura o are în funcția expeditorului, atunci semnătura digitală este validă dat fiind faptul că doar expeditorul prin intermediul cheii private are posibilitatea de a genera o semnătură validă.[21]

Exemplu:

Bob dorește să-i trimită un mesaj lui Alice și vrea ca mesajul să fie privat, disponibil doar pentru Alice. Astfel, Bob folosește o funcție securizată hash împreună cu cheia lui privată ca și intrare pentru algoritmul semnăturii digitale. Bob trimite mesajul împreună cu semnătura digitală atașată. În momentul în care Alice primește mesajul împreună cu semnătura, aceasta calculează valoarea hashului generat de către mesaj și folosește cheia publică pentru a verifica prin intermediul algoritmului semnătura digitală. Dacă în urma verificării valorile date sunt aceleași generate de către algoritmul lui Bob, semnătura este validă.

Datorită faptului că niciun alt utilizator nu are acces la cheia privată a lui Bob, astfel, nimeni nu ar fi putut să scrie un mesaj care să fie deschis cu cheia publică a acestuia, fiind asigurate și integritatea și autenticitatea mesajului.

3.7 Arhitectura unei infrastructuri bazată pe chei publice (PKIX)

Infrastructura bazată pe chei publice permite trimiterea mesajelor, datelor sau a oricăror informații de interes pentru utilizatori prin intermediul unui mediu nesigur cum ar fi internetul, astfel, tehnologia ajungînd să fie folosită pe scară largă. Unul dintre principalele sectoare în care este folosită infrastructura bazată pe chei publice este comerțul electronic, standardele pentru PKI fiind într-o continuă evoluție datorită complexității de care dau dovadă.[22]

The Internet Engineering Task sau prescurtat IETF reprezintă un grup de lucru care s-a axat pe înființarea unui model bazat pe X.509 care este potrivit pentru implementarea pe internet al unei arhitecturi care are la bază certificate [23].

Elementele Public Key Infrastructure X.509 sunt:

- Entitatea finală: reprezintă un termen folosit pentru a desemna utilizatorii finali ori orice alte entități care pot fi identificate în certificatul de chei publice.

- Autoritatea de Certificare: reprezintă emitentul certificatelor și, de obicei, lista de revocare a certificatului (CRLs). În unele cazuri pot deține diverse funcții administrative, deși, de cele mai multe ori acestea sunt diseminate către Autoritatea de Înregistrare.

- Autoritatea de înregistrare: reprezintă o componentă opțională care poate deține diverse funcții administrative preluare de la Autoritatea de Certificare.

- Emitentul CRL: reprezintă o componentă opțională prin intermediul căreia Autoritatea de Certificare o poate disemina către CRLs.

- Depozit: reprezintă un termen folosit generic care indică metoda de stocare a certificatelor cît și al CRLs astfel încît există posibilitatea să fie preluate de Entitățile finale.

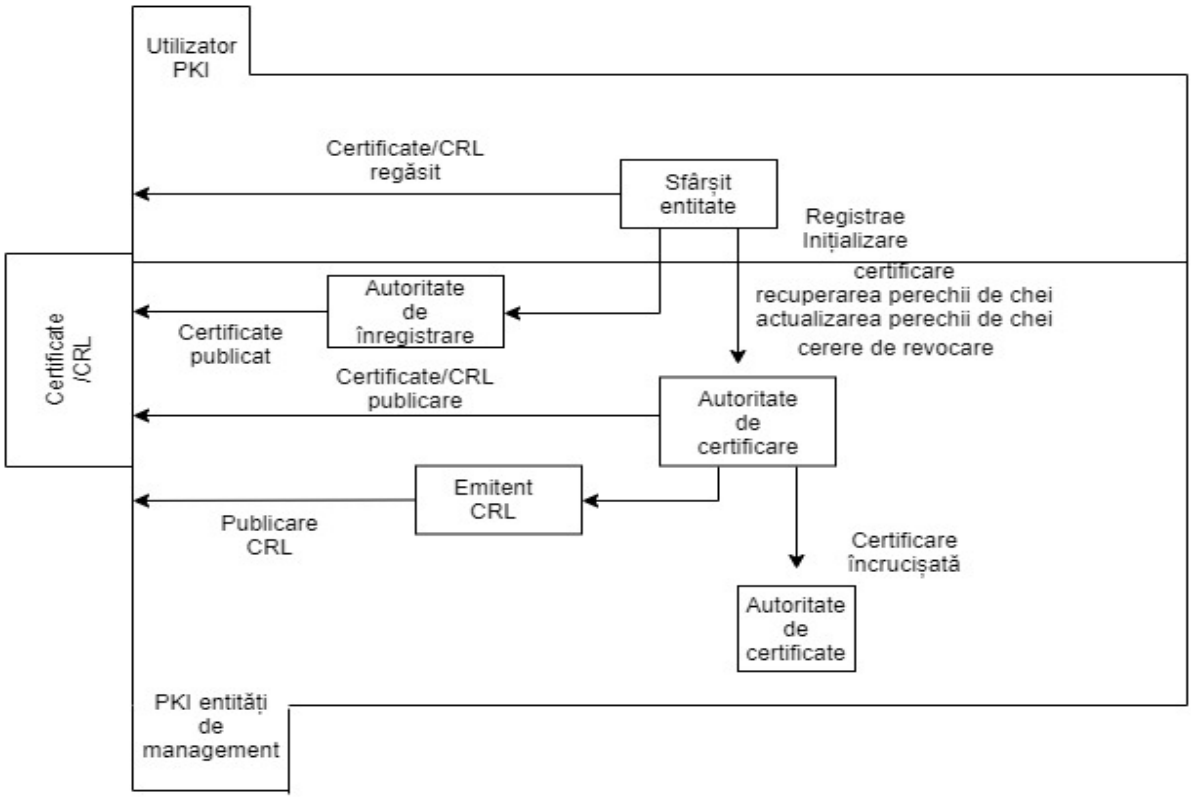

Fig.14-Elementele Public Key Infrastructure X.509 [1]

Public Key Infrastructure X.509 identifică un anumit număr de funcții care trebuie să fie susținute de anumite protocoale de management menționate în figura sus-menționată(Fig. 14), astfel:

- Registrare - acesta reprezintă un proces prin care un utilizator se identifică de către Autoritatea de certificare, direct sau prin intermediul autorității de înregistrare, prealabil autoritatea de certificare eliberînd un certificat pentru utilizatorul respectiv.
- Inițializarea - înainte ca sistemul clientului poate opera în siguranță este necesară instalarea materialelor care sunt în relație cu materialele aflate în infrastructură. De exemplu, utilizatorul trebuie să fie securizat prin intermediul cheii publice și alte informații din Autoritatea de certificare pentru validitate.
- Certificarea - acest proces include faptul că Autoritatea de certificare verifică cheia publică a utilizatorului, returnează certificatul respectiv către sistemul utilizatorului sau/și postează acest certificat în depozit.
- Recuperarea perechii de chei - perechile de chei pot fi folosite pentru a susține semnătura digitală, verificarea, criptarea și decriptarea sau ambele. Cînd o pereche de chei este folosită pentru criptare sau decriptare, este important să ofere un mecanism prin intermediul căruia să recupereze cheile respective necesare pentru decriptare în caz contrar recuperarea datelor fiind imposibilă. Recuperarea cheii permite astfel utilizatorilor să recupereze mesajele și informațiile. [24]
- Actualizarea perechilor de chei - toate perechile de chei trebuie să fie actualizate periodic sau înlocuite cu o nouă pereche de chei noi și eliberate noi certificate. Actualizarea este esențială în momentul în care durata de viață a certificatului a expirat și ca urmare a revocării certificatului.
- Cerere de revocare - utilizatorul autorizat care poate cere revocarea certificatului în cazul compromiterii cheii, numelui sau schimbări în afiliere.[25]
- Certificare încrucișată - acesta reprezintă un certificat emis de către o autoritate de certificare către alta care conține o cheie de semnături folosită pentru eliberarea certificatelor.

3.8 Rolul unei Autorităţi de Certificare

Autoritatea de Certificare reprezintă o instituţie care eliberează certificate digitale prin semnarea cererilor de certificare venite din partea utilizatorilor Autorităţii. Majoritatea Autorităţilor de certificare folosesc că schema criptografică PKI-ul, adică criptografia cu chei publice.

Autoritatea de Certificare reprezintă o terţă-parte în care două parţi diferite au încredere comună. În esenţă, Autoritatea certifica faptul că cel puţin o parte din cele două este autentică şi autorizată să desfăşoare operaţii criptografice în numele părţii respective.

Certificatele emise de către Autoritatea de Certificare dispun de cheia publică a utilizatorului, dar şi identitatea acestuia. Cheia publică este generată din cheia privată prin funcţii matematice. Cheia privată este secretă, aparţinînd şi fiind vizibilă doar proprietarului, Autoritatea de certificare ne fiind interesată de aceasta. Totodată, prin intermediul autorităţii se atestă faptul că utilizatorului îi aparţine cheia publică.[26] (Fig.16)

Compromiterea unei Autorităţi de Certificare reprezintă practic compromiterea securităţii oferită de această tuturor clienţilor care au folosit serviciile puse la dispoziţie de către autoritate şi au încredere în aceasta.

Pentru crearea unei imagini de ansamblu a dezastrului cauzat în urma combaterii unei Autorităţi de Certificare se dă următorul exemplu:

Se presupune că un atacator cu numele de Jon reuşeşte să obţină un certificat digital de la autoritate pe numele altei persoane, Georgia. Jon deţine cheia privată a Georgiei, astfel corelînd cu cheia publică din certificatul digital emis de către autoritate face că certificatul să pară real. Aşadar dacă Jon ar trimite un mesaj semnat cu acel certificat lui Bob, acesta ar crede că mesajul îi aparţine Georgiei. Dacă Bob i-ar răspunde Georgiei la mesajul presupus trimis de ea (de fapt trimis de Jon), el nu va şti că mesajul nu va fi citit de către aceasta şi că va avea acces la mesajul lui, Jon folosind cheia privată a Georgiei.

În anul 2001 o terţă parte a reuşit să compromită Autoritatea de Certificare Verifsign, putînd să convingă autoritatea la eliberarea a două certificate în numele companiei Microsoft, fiind emise pentru „Microsoft Corporation”. Din fericire aceasta fraudă a fost depistata rapid, iar Microsoft şi Autoritatea de Certificare Verisign au luat măsurile corespunzătoare.

Deoarece este destul de grea verificarea datelor de identitate prezentate unei Autorităţi de certificare cu entităţile care le prezintă, s-a luat măsura de a cere clienţilor care solicită serviciile autorităţii, anumite date personale (adresa fizică, locul de muncă, buletinul,etc.).

În unele arhitecturi mai mari PKI există posibilitatea lipsei de cunoştinţe a autorităţii care a emis certificate celeilalte părţi, de aceea există şansa ca şi Autoritatea de certificare a persoanei a doua să fie să conţină o cheie publică care este semnată de o altă autoritate comună celor două persoane. Astfel se poate verifica direct lanţul de încredere.(Fig.15)

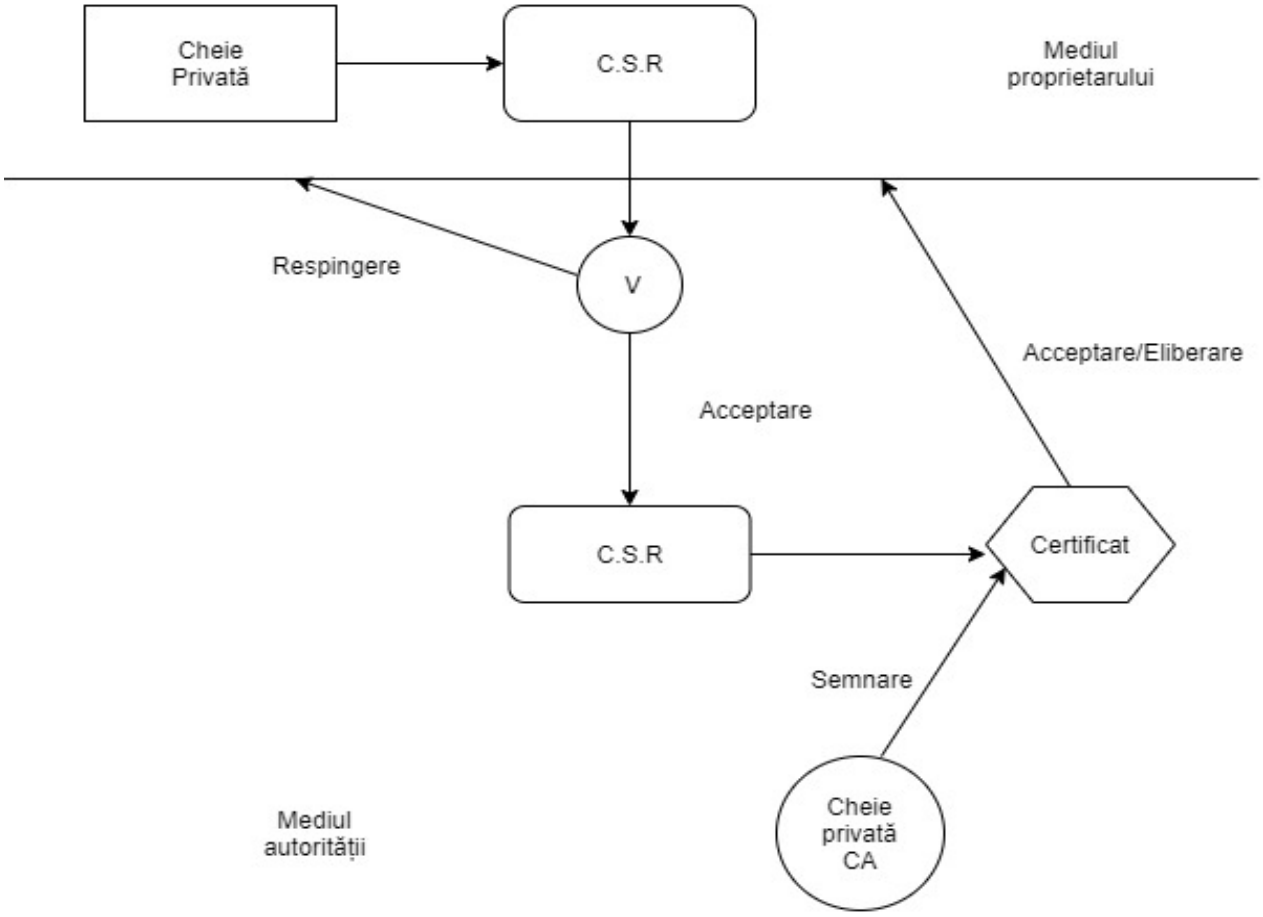

Fig.15-Etapele de parcurs pentru semnătura unui certificat de server de către o Autoritate

CAPITOL 4

AUTORITATI DE CERTIFICARE DIGITALA go&SIGN

În continuare, această lucrare prezintă conceptul de Autoritate de Certificare, implementat practic sub forma Autorităţii de Certificare "go&SIGN".

4.1. Design şi caracteristici

4.1.1. Caracteristici principale:

Aplicaţia aferenta acestei lucrări este „Autoritatea de certificare digitală go&SIGN". Autoritatea de certificare poate reprezenta atît o soluţie publică, globală, pentru emiterea de certificate (în caz că se rezolvă problema legală sau cea de încredere) cît şi o soluție internă, privată, pentru o anumită organizaţie, de creare a propriei autorităţi de certificare.

Autoritatea de Certificare „go&Sign" emite certificate digitale de client, viitorii clienţi ai Autorităţii avînd posibilitatea de a intra pe site şi de a-şi genera un certificat de client. Pentru emiterea de certificate client, nu este nevoie de vreo autentificare, certificatul putînd fi creat pe loc. Reînnoirea şi revocarea necesita, însă, introducerea unei parole (cea introdusă în cursul procesului de înregistrare) de către clientul autorităţii.

Aplicaţia reprezintă un portal web dinamic şi interactiv care conţine un centru de certificare pentru certificatele de client şi o secţiune de documentaţie care prezintă mai multe tutoriale tehnice de implementare.

4.1.2. Funcţionalităţi:

Majoritatea autorităţilor de certificare au de urmat o întreagă serie de proceduri pentru a asigura buna calitate a serviciilor oferite clienţilor. Principalele responsabilităţi ale unei A.C. sunt:

- emiterea de certificate sau respingerea de certificate
- revocarea de certificate
- reînnoirea de certificate
- semnarea de obiecte

Autoritatea de certificare păstrează, în baza ei de date, în mod securizat, toate informaţiile legate de clienţii acesteia şi de certificatele emise.

Mai jos, această lucrare detaliază principalele responsabilităţi ale unei A.C.:

Emiterea de certificate – rolul primordial al unei Autorităţi de Certificare

• Pasul 1: Un nou utilizator care doreşte să obţină un Certificat de client intra pe site-ul A.C. şi se înregistrează pe site cu informaţiile personale

• Pasul 2: După completarea informaţiilor personale şi aplicarea pentru un certificat nou, A.C. efectuează anumite verificări procedurale pentru completarea corectă a formularului, verificarea duplicităţii înregistrărilor şi a validităţii acestora.

• Pasul 3: Dacă toate informaţiile sunt corecte şi validarea datelor trece cu succes, atunci A.C. efectuează următoarele proceduri de securitate în vederea eliberării certificatului de client:

- Generează cheia privată a clientului;
- Criptează cheia private a clientului folosind parola furnizată în formularul de înregistrare;
- Generează cererea de certificare (CSR) (care conţine cheia publică aferenta cheii private mai sus menţionate) care conţine anumite informaţii de bază furnizate de client în formularul de înregistrare;
- Semnează cererea de certificare (CSR) cu Cheia Privată a A.C. pentru a rezulta un Certificat Final de Client (CRT)
- Certificatul Final de client, mai apoi, este combinat, în format PKCS#11, cu cheia privată, rezultînd, în final, un fişier binar (non-ASCII) P12 (*.p12) care este protejat prin parola (criptat).
- Mai apoi, acest Certificat de Client (format P12) poate fi utilizat în 2 moduri:
- Instalat pe un Token criptografic, dacă clientul deţine unul (procedura de instalare pe token diferă de la situaţie la situaţie)
- Instalat pe calculatorul local (în store-ul specific)

• Pasul 4: A.C. reţine, în baza ei de date, certificatul clientului, împreună cu cheia privată criptata cu parola introdusă la înregistrare, iar parola este ţinută sub formă de Suma de Control, pentru a asigura că doar Clientul poate efectua operaţiuni criptografice, dar şi facilitînd, în acelaşi timp, A.C. în a reînnoi certificatul sau a efectua anumite operaţiuni criptografice cu acesta, pe platformă A.C. (ex: Semnarea de documente online).

Respingerea de certificate – Autoritatea de Certificare are dreptul de a refuza cererile de certificare dacă detectează anumite probleme la înregistrare sau pentru alte motive întemeiate.

Revocarea de certificate – Această funcţie se utilizează de obicei atunci cînd proprietarul certificatului doreşte să nu mai beneficieze de certificat, din diferite motive. Clientul autorităţii accesează site-ul autorităţii şi solicită revocarea certificatului digital, iar în urma solicitării, A.C. îi va retrage certificatul şi îl va publica în lista certificatelor revocate. Din acel moment orice navigator care caută starea acelui certificat retras va primi avertizarea că certificatul a fost revocat de către Autoritatea de Certificare.

Totodată, autoritatea este capabilă de a revoca forţat un certificat în cazul unei suspiciuni de ilegalitate (de exemplu folosirea unui certificat în scopul unor atacuri).

Reînnoirea de certificate – Certificatele au un termen de valabilitate, denotat de două cîmpuri din structura unui certificat X509. Certificatul poate fi utilizat doar în perioada în care este valabil. În cazul în care certificatul expira iar utilizatorul doreşte să mai beneficieze de el, atunci acestuia i se permite posibilitatea de reînnoire. Autoritatea de Certificare preia Cererea de Certificare (CSR-ul) salvat în baza de date de la prima semnare şi îl resemnează folosind cheia privată a autorităţii, rezultînd un certificat nou cu un alt termen de valabilitate.

Semnarea de obiecte – Autoritatea de Certificare oferă posibilitatea de a semna documente cu condiţia deţinerii de certificat digital emis de către autoritate. Utilizatorului îi se va cere, pe lîngă ataşarea obiectului pe care vrea să îl semneze, CNP-ul şi parola pe care a utilizat-o, astfel obţinînd semnătură electronică.

Certificatele de client îşi găsesc utilitatea în autentificarea la o resursă cu regim special de acces sau pentru a semna documente electronic.

În acest sens a fost inventat standardul PKCS folosit în mod special pentru certificatele digitale de client. Certificatele digitale în format PKCS conţin partea publică a certificatului şi cheia privată, putînd fi importate în navigatoare din computere personale dar şi pe dispozitive speciale criptografice (token-uri).

Unul dintre dispozitivele criptografice speciale este Token-ul, construit pe conectivitate USB, asemănîndu-se cu stick-urile de memorie USB. Spre deosebire de stick-urile de momorie care stochează informaţii, token-ul stochează chei, parole şi elemente criptografice de securitate informatică.

Certificatele digitale de client generate de către Autoritatea de Certificare go&SIGN se pot încărca pe token iar ulterior se pot folosi în scopul autentificării pe un anumit site cu

regim special, creat pentru partea de testare. De asemenea, certificatele digitale de client pot fi folosite şi pentru semnarea electronică de documente.

În scopul testării certificatelor de client emise de către Autoritate, s-a generat un site secundar (modul secundar) denumit în continuare Catalogul de Note. Acest catalog reprezintă o aplicaţie web care acceptă autentificarea cu certificat digital de client pentru autentificarea cadrelor didactice.

Aplicaţia de Catalog are două tipuri de utilizatori: profesor şi student. De asemenea, întîlnim tot două tipuri de autentificare, în funcţie de gradul utilizatorului: profesor sau student. Astfel:

- Pentru profesor este necesar uşerul, parola şi certificatul digital de client luat de pe token;
- Pentru student trebuie doar uşerul şi parola;

Partea aceasta aplicativă de testare are două funcţii primordiale, şi anume adăugarea de note şi vizualizarea acestora. Profesorul, după autentificarea executată corect, cu certificatul înserat de pe token, are posibilitatea de a vizualiza lista cu studenţii care au note la disciplină predată de acesta, putînd să editeze sau să şteargă cîmpul respectiv. De asemenea, profesorul poate să adauge note studenţilor.

Studentul, după introducerea numelui de utilizator şi a parolei, este redirecţionat către pagina principală care-i afişează o listă cu notele pe care acesta le-a obţinut în decursul anului, la materiile pe care le studiază.

4.2 Arhitectura aplicaţiei

Diagrama use-case din fig x care descrie secvenţele de acţiuni pe care aplicaţia le executa atunci cînd interacţionează cu entităţi implicate (clienţii Autorităţii de Certificare go&SIGN) şi care duc la managementul certificatului solicitat şi utilizat.(Fig. 16)

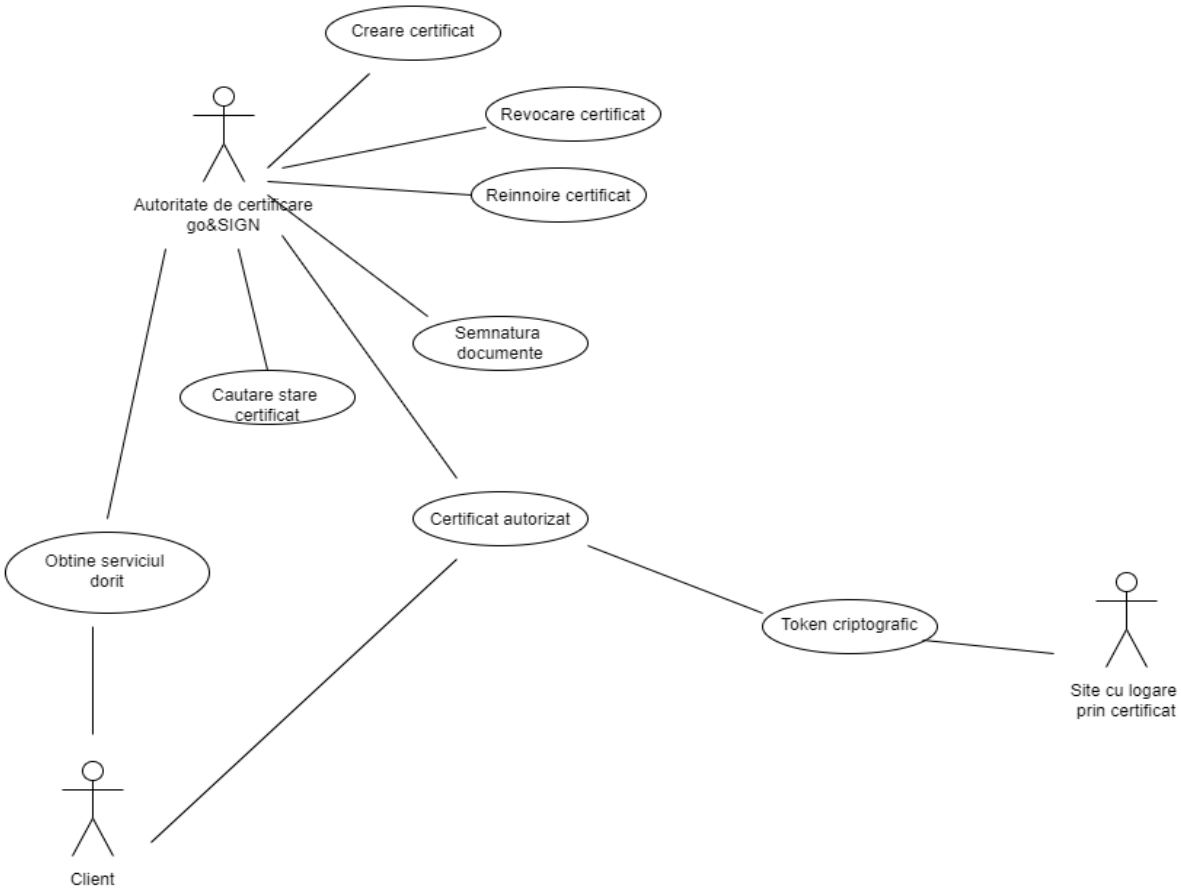

Fig. 16- Diagrama USE-CASE care descrie aplicatia

Partea arhitecturală cuprinde:

- Modulele aplicaţiei:
- Platforma Web de certificare (Certificate de Client)
- Componentă de semnare electronică (Semnătură electronică)
- Componentă de verificare semnătură electronică
- Aplicaţie de testare pentru autentificarea cu certificat digital (Catalog)
- eToken criptografic – pentru stocat certificate digitale de client şi folosit pentru autentificare

Interacţiunea dintre module:

- Platforma de Certificare cu Aplicaţia de semnare electronică: interacţionează prin faptul că platformă de certificare stochează securizat, în baza de date, certificatul clientului, care este folosit de aplicaţia de semnare.

Elemente comune: Tehnologii Web, MySQL.

- Platforma de Certificare cu Aplicaţia de verificare: Nu exista interacţiuni.

  Elemente comune: nu sunt.

- Platforma de Certificare cu Catalogul: Platforma de certificare nu are o legătură directă cu catalogul, însă are o legătură indirectă prin faptul că furnizează Certificatul de Client necesar pentru autentificarea pe catalog.

  Elemente comune: Certificatul de Client.

- Aplicaţia de semnare electronică cu Aplicaţia de verificare electronică: nu are o legătură directă, însă există o legătură indirectă prin faptul că elementul comun, în cazul ambelor este reprezentat de Semnătură electronică

- Aplicaţia de semnare electronică cu Catalogul: nu există nicio legătură directă, însă există o legătură indirectă prin faptul că elementul comun, în cazul ambelor, este reprezentat de Certificatul de Client.

- Aplicaţia de verificare a semnăturii cu Catalogul: nu există nicio legătură, de niciun fel.

  În această diagramă se prezintă scenariul în care unul dintre clienţii AC doreşte să obţină o semnătură digitală pe un document propriu prin intermediul Autorităţii de Certificare. (Fig.17)

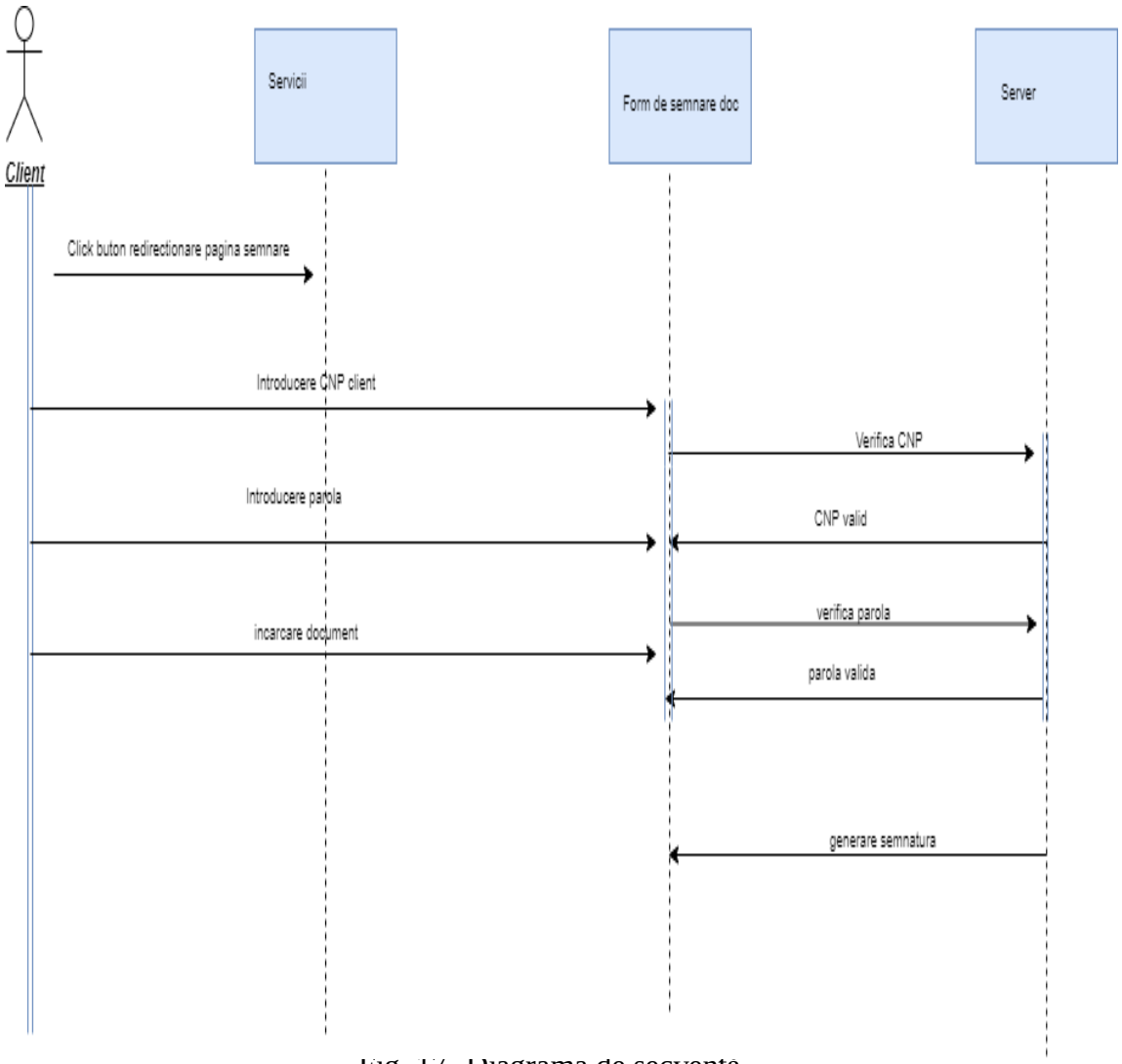

Fig. 17 -Diagrama de secvență

După accesarea aplicaţiei „go&SIGN", clientul navighează la „Servicii" unde îşi caută serviciul la care vrea să apeleze. Găsind opţiunea de semnare, utilizatorul încarcă documentul şi completează cîmpurile cu CNP-ul (pentru că autoritatea să verifice în baza ei de date dacă clientul deţine certificat digital creat de aceasta) şi parolă cu care acesta şi-a generat certificatul digital. După verificarea deţinerii unui certificat digital emis de către autoritate, aceasta semnează documentul încărcat de către client cu cheia privată a lui.

Prin diagrama de activităţi (Fig. 18) se explica că:

- Se poate reîncerca procedura de creare a unui certificat digital dacă eşuează generarea.
- După emiterea unui certificat digital, utilizatorului îi sunt afişate obiectele criptografice pentru incluziune pe Token, şi în acelaşi timp sunt stocate în baza de date (cheia privată criptata şi informaţiile de uz public)

- În căutarea unui client al autorităţii, se verifică în baza de date a acesteia şi se returnează un răspuns afirmativ (dacă deţine certificat digital de către autoritate) sau negativ (dacă nu deţine certificat digital de către autoritate)
- În cazul revocării unui certificat digital,înregistrarea din baza de date referitoare la client este ştearsă
- În ceea ce priveşte reînnoirea,cheia privată este decriptată cu parola introdusă cît pentru a genera CSR-ul, iar mai apoi obiectele criptografice, în același timp se trece valabilitatea nouă în baza de date;

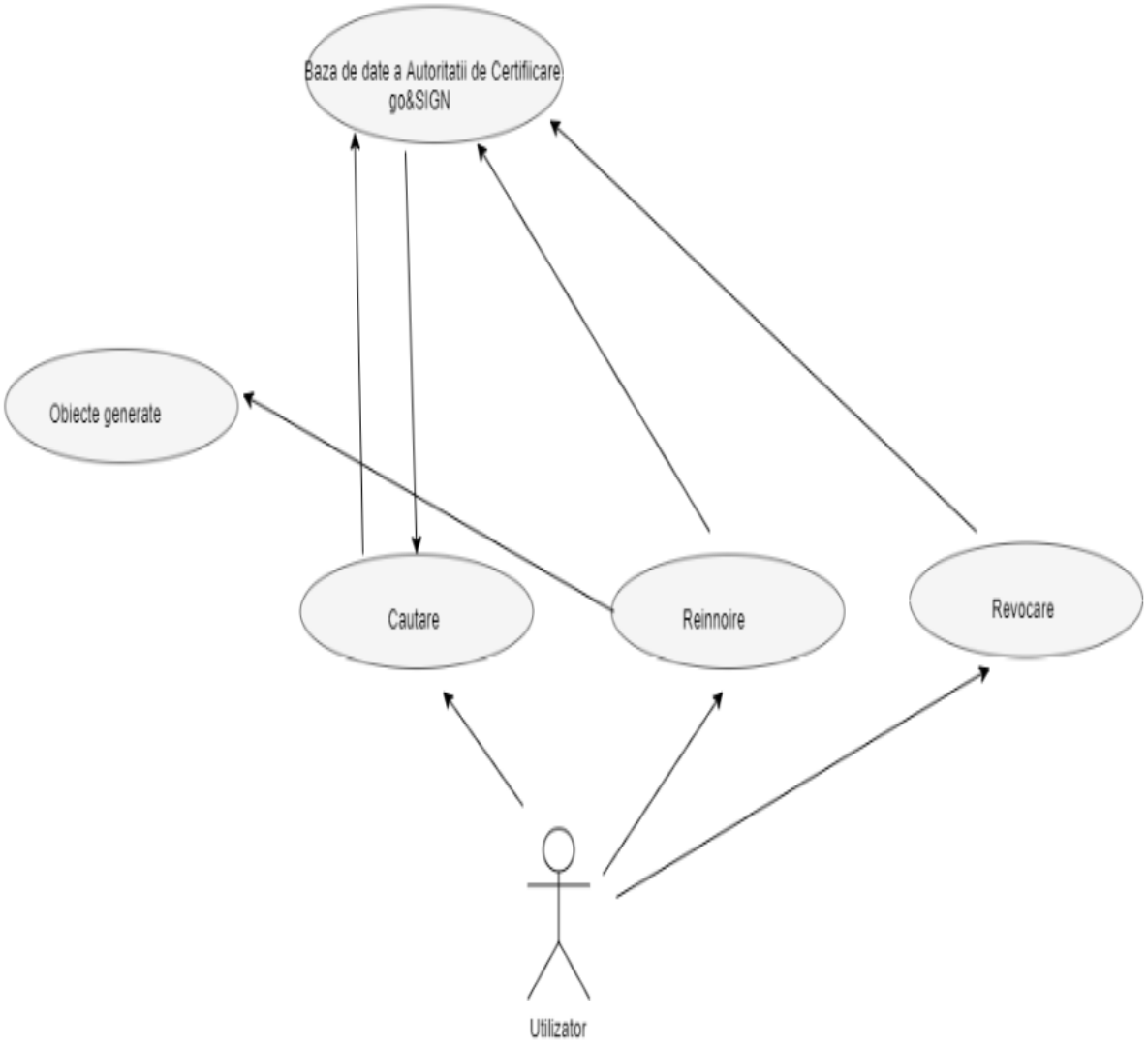

Fig.18-Diagrama de activitaţi

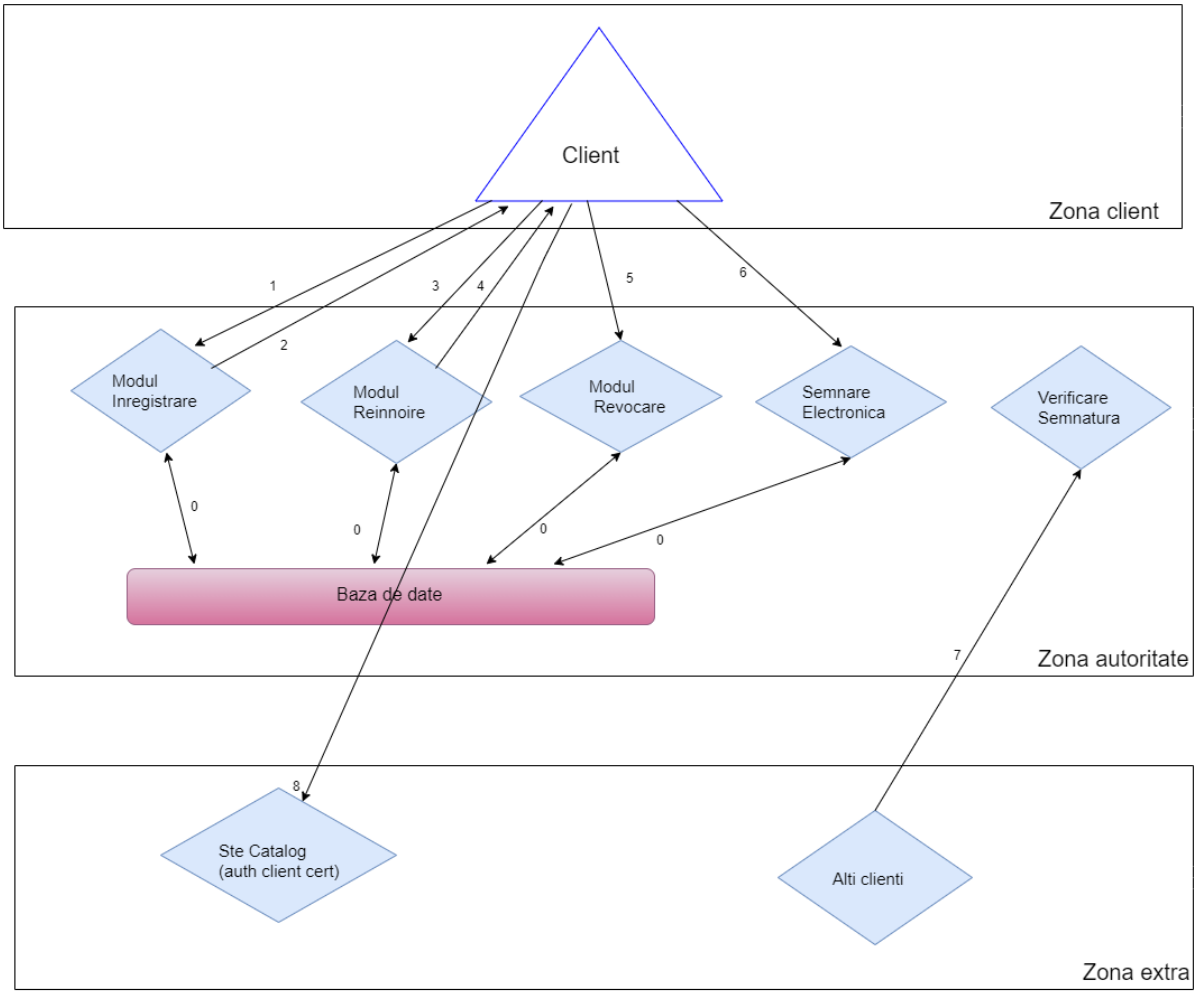

Fig. 19- Diagrama DSD

Legăturile diagramei (Fig. 19) :

0 → Legătură permanentă, bidirecţională, a autorităţii cu baza ei de date în vederea procesării informaţiilor

1 → Înregistrare pt obţinere certificat nou

2 → Autoritatea oferă noul certificat

3 → Cerere reînnoire

4 → Rezultat reînnoire (eliberare certificat reînnoit)

5 → Cerere revocare

6 → Cerere de semnare electronică a unui document

7 → Alţi clienţi verifica obiectul de semnat, cu cheia publică, dacă într-adevăr aparţine cheii private

8 → Clientul se autentifica în zona externă pe un alt site (de test) cu certificatul obţinut de la Autoritate

4.3 Tehnologii

În cadrul acestui subcapitol se vor prezenta tehnologiile de programare cu care funcţionează aplicaţia şi de care componentele acesteia au nevoie pentru a funcţiona interconectat.

Găzduire web cu Apache: tehnologia aceasta permite accesarea site-ului de către public prin intermediul găzduirii acestuia pe internet, punîndu-l la dispoziţia oricui.

Serverul Apache suporta conexiuni multiple în paralel şi oferă un echilibru între calitate şi numărul de vizitatori.

Fiind o autoritate proaspăt infintata, go&SIGN nu are cheile publice incluse în marile navigatoare, de aceea trebuie instalat certificatul rădăcina al Autorităţii de Certificare care conţine şi cheia publică înainte de a intra pe paginile securizate, altfel utilizatorului îi se va afişa o atenţionare la întîlnirea certificatului pe server pentru portalul dinamic al Autorităţii de Certificare.

Baza de daate cu MySQL: folosită în scopul stocării datelor referitoare la utilizatorii Autorităţii de Certificare go&SIGN, cererile de certificare, certificatele digitale ale clienţilor şi alte informaţii.

Scripting dinamic cu HTML şi PHP: aceste tehnologii sunt cele mai utilizate în toată aplicaţia, HTML-ul fiind folosit pe partea de design, iar PHP-ul carnd motoarele interne ale aplicaţiei.

Trimiterea de mail cu Sendmail: tehnologie responsabilă cu trimiterea de mesaje prin poştă electronică a clientului. După fiecare utilizare a Autorităţii de Certificare, în funcţie de serviciile la care apelează clientul, acesta va primi cîte-un mesaj (mesaj cînd o cerere de a fost acceptată sau respinsă, mesaj cînd un certificat expira sau este revocat etc.). Chiar dacă pe portal îi sunt afişate aceste informaţii clientului, pentru o mai buna organizare şi aducere la cunoştinţă a acestuia, datele îi sunt transmise şi printr-un mesaj pe adresa lui de mail. În acest fel, clientul Autorităţii observa şi pe mail, prin prisma mesajului trimis de către aplicaţie, că s-au efectuat operaţii în cadrul Autorităţii de Certificare sub numele lui. Toate aceste mesaje de informare sunt transmise de către Autoritate prin prisma completării unui cîmp cu adresa personală de e-mail de către utilizator.

În ceea ce priveşte necesitatea hardware a clientului, acesta are nevoie de un navigator web compatibil HTML 4.0 sau XHTML 1.0 (Opera, Google Chrome, Microsoft Edge) şi de un mail al clientului.

Pentru asigurarea calităților operaţiilor criptografice este nevoie de un Token criptografic (Alladin eToken) pentru încărcarea obiectelor de certificare pe acesta dar şi de un client token PKI pentru operaţiile efectuate în mod manual pe token.

Baza de date a Autorităţii de certificare go&SIGN este de tip MySQL şi conţine mai multe tabele relaţionale care cuprind informaţii cu privire la clienţii autorităţii şi la certificatele digitale ale acestora, precum şi alte cîteva informaţii de control şi management.

În tabela „pkcs" sunt stocate datele personale ale clienţilor, care se împart în două categorii:

- Date generale: nume-prenume, CNP, domiciliu, tara, adresa de email
- Date private sensibile: cheia privată criptată (cu parola) şi parola în formă de suma de control.

Pentru a proteja datele private sensibile de eventuale atacuri sau accesări neautorizate, chiar şi la nivelul Autorităţii de Certificare, Cheia Privată este criptata cu Parola clientului sub o formă de Criptare Simetrică, iar Parola propriu-zisă este sub formă de Suma de Control pentru a nu putea recupera sub nicio formă parola propriu-zisă folosită la Criptarea Cheii Private

Pentru a ţine o evidenţă a stării certificatelor digitale, în baza de date se regăseşte statutul (sau codul de stare) fiecăruia, astfel:

- Pentru un certificat digital de client emis cu succes şi aflat în perioada de valabilitate, acesta are codul de stare: „A" - acceptat;
- Pentru un certificat digital de client căruia îi este expirat termenul de valabilitate i se atribuie codul de stare „E" - expirat;
- Pentru un certificat de client care a fost revocat (şters) de deţinătorul acestuia sau de către A.C. din motive întemeiate, acestuia i se atribuie codul de stare „G" - revocat;

CAPITOLUL 5

CONCLUZII

Tehnologiile de securitate ale informaţiei au un imens impact asupra modului în care sunt gestionate în ziua de azi datele digitale. Din acest punct, de vedere platformele PKI îşi găsesc o eficienţă deosebit de ridicată, avînd un dublu rol: criptare şi semnare digitale.

Lucrarea implementează o Autoritate de Certificare (AC) atît din punct de vedere tehnic cît şi din punct de vedere procedural. Algoritmii de criptare utilizaţi se regăsesc în cadrul implementărilor reale ale AC.

Din punct de vedere practic soluţia prezentată în lucrare poate constitui o platformă PKI a unei organizaţii.

BIBLIOGRAFIE

[1] CRYPTOGRAPHY AND NETWORK SECURITY PRINCIPLES AND PRACTICE SEVENTH EDITION GLOBAL EDITION William Stallings Boston – informații extrase în data de 19 Mai 2021

[2] http://www.management.ase.ro/reveconomia/2004-special2/20.pdf – informații extrase în data de 24 Mai 2021

[3] https://sectigostore.com/blog/what-is-asymmetric-encryption-how-does-it-work/ – informații extrase în data de 28 Mai 2021

[4]https://www.scientia.ro/tehnologie/34-cum-functioneaza-calculatorul/409-cum-functioneaza-criptarea-datelor.html – informații extrase în data de 1 Iunie 2021

[5] https://virtual-academy.ro/index.php/resurse/53-elemente-de-criptografie-5–informații extrase în data de 10 Iunie 2021

[6] http://ares.utcluj.ro/tsi/tsi/index_files/Cap3.pdf – informații extrase în data 12 Iunie 2021

[7] https://www.tutorialspoint.com/cryptography/advanced_encryption_standard.htm – informații extrase în data de 17 Iunie 2021

[8] https://www.educative.io/edpresso/what-is-the-aes-algorithm – informații extrase în data de 19 Iunie 2021

[9] https://cheapsslsecurity.com/blog/what-is-asymmetric-encryption-understand-with-simple-examples/ – informații extrase în data de 20 Iunie 2021

[10] http://www.aut.upt.ro/~bgroza/Slides/Carte_Intro_Cripto.pdf – informații extrase în data de 22 Iunie2021

[11]https://www.proiecte.ro/calculatoare/criptografie-cu-chei-publice-16631-informații extrase în data de 24 Iunie 2021

[12] https://www.di-mgt.com.au/rsa_alg.html -informații extrase în data de 24 Iunie 2021

[13]https://www.tutorialspoint.com/cryptography_with_python/cryptography_with_python_understanding_rsa_algorithm.htm- informații extrase în data de 24 Iunie 2021

[14] http://ruxandraolimid.weebly.com/uploads/2/0/1/0/20109229/crypto_c_22.pdf-informații extrase în data de 24 Iunie 2021

[15] http://en.wikipedia.org/wiki/Cryptographic_hash_function - informații extrase în data de 24 Iunie 2021

[16] https://ro.eyewated.com/functia-de-criptare-criptografica/- informații extrase în data de 26 Iunie 2021

[17] https://staff.fmi.uvt.ro/~stelian.mihalas/cri_sin/cursuri/crisin.pdf-informații extrase în data de 27 Iunie 2021

[18] https://www.colorful.hr/semnatura-electronica-ce-este-si-cum-o-poti-obtine/-informații extrase în data de 27 Iunie 2021

[19] https://heritage-offshore.com/securitatea-informaiilor/ce-sunt-semnturile-digitale-i-cum-funcioneaz/- informații extrase în data de 27 Iunie 2021

[20] https://www.geeksforgeeks.org/digital-signature-standard-dss/- informații extrase în data de 27 Iunie 2021

[21] https://www.cs.kau.se/cs/dvgc19/ch13.pdf -informații extrase în data de 28 Iunie 2021

[22]http://www.marketwatch.ro/articol/2696/Tehnologia_PKI__cheia_integritatii_informatiei/ -informații extrase în data de 28 Iunie 2021

[23] https://datatracker.ietf.org/wg/pkix/about/- informații extrase în data de 28 Iunie 2021

[24]https://www.hyperelliptic.org/tanja/teaching/cryptoI13/cryptodict.pdf?fbclid=IwAR2gk7tMQfQPmbSsXNjKPh09JeadHt6N0ZtJ9mKzFNjVU8oV8LMxgsMyk7o- informații extrase în data de 29 Iunie 2021

[25]https://crypto.stanford.edu/~dabo/cryptobook/draft_0_2.pdf?fbclid=IwAR1caMvikQkHyEXKkqaUs47WWspsFsWO7OCfCAL9HV8Vpdvfj_kGomXVKeo- informații extrase în data de 30 Iunie 2021

[26] https://www.ibm.com/docs/ro/i/7.1?topic=concepts-certificate-authority- informații extrase în data de 30 Iunie 2021