Algoritmi RSA

Erjon Asllani
Universiteti i Prishtinës
Fakulteti i Shkencave Matematike-Natyrore
Shkenca Kompjuterike
Prishtinë, Kosovë
erjon.asllani@student.uni-pr.edu

Fatbardh Gashi
Universiteti i Prishtinës
Fakulteti i Shkencave Matematike-Natyrore
Shkenca Kompjuterike
Prishtinë, Kosovë
fatbardh.gashi6@student.uni-pr.edu

Vullnet Gërvalla
Universiteti i Prishtinës
Fakulteti i Shkencave Matematike-Natyrore
Shkenca Kompjuterike
Prishtinë, Kosovë
vullnet.gervalla@student.uni-pr.edu

I. Hyrje

Algoritmi RSA (Rivest-Shamir-Adleman) është një algoritëm enkriptimi dhe dekriptimi i përdorur gjerësisht që revolucionarizoi fushën e kriptografisë. Ai u prezantua për herë të parë në 1977 nga Ron Rivest, Adi Shamir dhe Leonard Adleman në Institutin e Teknologjisë në Massachusetts (MIT). Algoritmi RSA ofron një metodë të sigurt dhe efikase për sigurimin e konfidencialitetit, integritetit dhe autenticitetit të transmetimit të të dhënave përmes rrjeteve të pasigurta. Në këtë punim, ne do të eksplorojmë motivimet pas krijimit të RSA, të metat e algoritmeve të mëparshme kriptografike dhe se si RSA i adresoi këto kufizime.

***Fjalët kyçe*—Kriptografia, Algoritmi RSA, Algoritmet asimetrike**

II. Algoritmi RSA

Përpara zhvillimit të RSA (Rivest-Shamir-Adleman), teknikat tradicionale kriptografike mbështeteshin shumë në algoritmet simetrike. Këto algoritme përfshinin përdorimin e të njëjtit çelës për proceset e enkriptimit dhe dekriptimit. Ndërsa algoritmet e çelësave simetrik ishin efikas dhe siguronin një nivel të lartë sigurie, ata vuanin nga një e metë themelore - shpërndarja e çelësave. Për të krijuar një komunikim të sigurt midis dy palëve, kërkohej një kanal i sigurt për të shkëmbyer çelësin e përbashkët. Ky proces ishte shpesh i prekshëm ndaj përgjimeve dhe ndërhyrjeve, duke rrezikuar sigurinë e sistemit.

Një tjetër pengesë e rëndësishme e algoritmeve simetrike ishte çështja e shkallëzueshmërisë. Me rritjen e numrit të përdoruesve ose pjesëmarrësve në një rrjet të sigurt komunikimi, numri i çelësave të kërkuar për komunikim të sigurt rritet në mënyrë eksponenciale. Kjo rritje eksponenciale e bëri menaxhimin e çelësave një proces jashtëzakonisht kompleks dhe të rëndë, që shpesh çon në kompromise kryesore dhe shkelje të sigurisë.

Për të kapërcyer këto kufizime, algoritmi RSA u zhvillua bazuar në konceptin e kriptografisë asimetrike. Kriptografia me çelës asimetrik, e njohur gjithashtu si kriptografia me çelës publik, përdor një palë çelësa - një çelës publik për kriptim dhe një çelës privat për dekriptim. Çelësi publik shpërndahet dhe mund të aksesohet lirisht nga kushdo, ndërsa çelësi privat mbetet konfidencial dhe i njohur vetëm për marrësin e synuar.

Motivimi pas krijimit të RSA ishte adresimi i problemit të shpërndarjes së çelësit te algoritmet simetrike. Me RSA, çelësi publik mund të ndahet lirisht dhe të përdoret nga kushdo për të kriptuar mesazhet, duke eliminuar nevojën për një mekanizëm të sigurt shkëmbimi të çelësave. Mesazhi i koduar mund të deshifrohej vetëm duke përdorur çelësin privat përkatës, i cili mbeti i sigurt tek marrësi i synuar. Kjo risi ofroi një zgjidhje praktike për komunikim të sigurt në një mjedis të hapur rrjeti.

Një nga avantazhet kryesore të RSA mbi algoritmet simetrike ishte aftësia e tij për të vendosur komunikim të sigurt midis dy palëve që nuk kishin asnjë ndërveprim paraprak ose sekrete të përbashkëta. Kjo veçori e bëri RSA veçanërisht të dobishme për komunikim të sigurt përmes internetit, ku përdoruesit mund të enkriptonin lehtësisht mesazhet duke përdorur çelësin publik të marrësit pa pasur nevojë për ndonjë marrëveshje paraprake.

Për më tepër, algoritmi RSA ofroi gjithashtu një zgjidhje efektive për problemin e menaxhimit të çelësave. Në një rrjet që përdor enkriptimin RSA, çdo pjesëmarrës ka një çift çelësash unik të përbërë nga një çelës publik dhe një çelës privat. Çelësat

publikë mund të ndaheshin lirisht, ndërsa çelësat privatë mbeten të ruajtur në mënyrë të sigurt. Kjo eliminoi nevojën për një sistem të centralizuar të menaxhimit të çelësave, duke reduktuar kompleksitetin e shpërndarjes së çelësave dhe duke minimizuar rrezikun e vjedhjes së çelësit.

Gjithashtu, RSA siguroi një nivel më të lartë sigurie në krahasim me algoritmet simetrike. Siguria e RSA bazohet në kompleksitetin matematikor të faktorizimit të numrave të mëdhenj të thjeshtë. Vështirësia e faktorizimit të numrave të mëdhenj përbën themelin e algoritmit RSA. Ndërkohë që është e mundur nga pikëpamja llogaritëse të shumëzohen dy numra të mëdhenj të thjeshtë për të gjeneruar çelësat publikë dhe privatë, faktorizimi i produktit të këtyre numrave të thjeshtë në numrat e parë origjinal besohet të jetë i pamundur nga pikëpamja llogaritëse, madje edhe me kompjuterët më të fuqishëm të disponueshëm sot. Ky kompleksitet matematikor ofron një nivel të fortë sigurie kundër sulmeve që tentojnë të përcaktojnë çelësin privat nga çelësi publik.

A. Aplikimet e RSA

Paraqitja e algoritmit RSA pati një ndikim të rëndësishëm në fusha dhe aplikime të ndryshme. Një fushë që përfitoi shumë nga RSA ishte komunikimi i sigurt përmes internetit. Para RSA, komunikimi i sigurt midis dy palëve përmes një rrjeti të hapur si interneti ishte një detyrë sfiduese. Algoritmi RSA bëri të mundur që individët dhe organizatat të shkëmbejnë në mënyrë të sigurt informacione të ndjeshme, të tilla si transaksionet financiare, të dhënat personale dhe dokumentet konfidenciale, në internet, pa cenuar sigurinë e tyre.

Një tjetër aplikim i dukshëm i RSA është në nënshkrimet dixhitale. Nënshkrimet dixhitale ofrojnë një mjet për të verifikuar autenticitetin dhe integritetin e dokumenteve ose mesazheve elektronike. Duke përdorur RSA, një dërgues mund të enkriptojë një vlerë hash të dokumentit duke përdorur çelësin e tij privat, duke krijuar një nënshkrim dixhital. Më pas, marrësi mund të deshifrojë nënshkrimin duke përdorur çelësin publik të dërguesit dhe ta krahasojë atë me vlerën e llogaritur hash të dokumentit të marrë. Nëse të dyja përputhen, kjo tregon se dokumenti nuk është manipuluar dhe se dërguesi është autentik. Ky përdorim i RSA në nënshkrimet dixhitale është bërë një komponent jetik në sigurimin e integritetit të informacionit dixhital.

Ndikimi i algoritmit RSA shkon përtej komunikimit të sigurt dhe nënshkrimeve dixhitale. Ai ka gjetur aplikime në fusha të tjera të ndryshme, si komunikimi i sigurt me email, protokollet e sigurta të aksesit në distancë, ruajtja e sigurt e të dhënave dhe sistemet e sigurta të votimit elektronik. Shkathtësia dhe efektiviteti i algoritmit RSA e kanë bërë atë një mjet themelor në kriptografinë moderne.

B. Kufizimet e RSA

Pavarësisht nga avantazhet e tij të shumta, algoritmi RSA nuk është pa kufizime. Një nga shqetësimet kryesore me RSA është kostoja e tij e përgjithshme llogaritëse. Operacionet RSA, veçanërisht fuqizimi modular, mund të jenë intensive nga pikëpamja llogaritëse, veçanërisht kur kemi të bëjmë me madhësi të mëdha të çelësave. Kjo ngarkesë llogaritëse mund të ndikojë në performancën e sistemeve që mbështeten shumë në enkriptimin dhe dekriptimin me RSA. Për të adresuar këtë çështje, janë propozuar optimizime dhe përmirësime të ndryshme, si përdorimi i algoritmeve efikase për fuqizimin modular dhe zhvillimi i përshpejtuesve harduerikë të krijuar posaçërisht për llogaritjet RSA.

Një sfidë tjetër e lidhur me RSA është cenueshmëria ndaj disa sulmeve, të tilla si sulmet e kohës, sulmet e kanalit anësor dhe sulmet e zgjedhura të tekstit të koduar. Sulmet e kohës shfrytëzojnë ndryshimet në kohën e ekzekutimit të operacioneve të ndryshme kriptografike për të nxjerrë informacione rreth çelësit privat. Sulmet e kanalit anësor analizojnë informacionin e rrjedhur përmes karakteristikave fizike të zbatimit, të tilla si konsumi i energjisë ose rrezatimi elektromagnetik, për të nxjerrë çelësin privat. Sulmet e tekstit të koduar përfshijnë një palë të jashtme që merr dekriptimin e teksteve të koduara dhe përdorimin e këtij informacioni për të nxjerrë çelësin privat. Këto sulme theksojnë rëndësinë e zbatimit të RSA në mënyrë korrekte dhe të sigurt për të zbutur dobësitë e mundshme.

Në vitet e fundit, përparimet në fuqinë kompjuterike, veçanërisht zhvillimi i kompjuterëve kuantikë, kanë paraqitur një kërcënim potencial për

RSA dhe algoritme të tjera tradicionale kriptografike me çelës publik. Kompjuterët kuantikë kanë potencialin për të zgjidhur problemin e faktorizimit të numrave të plotë, i cili është në thelb të sigurisë së RSA. Si rezultat, ka pasur një interes në rritje për zhvillimin dhe kalimin në algoritme kriptografike post-kuantike që mund t'i rezistojnë sulmeve nga kompjuterët kuantikë. Studiuesit po eksplorojnë në mënyrë aktive alternativa ndaj RSA, të tilla si kriptografia e "lattice-based", kriptografia e bazuar në kod, kriptografia me shumë variacione dhe nënshkrimet dixhitale të bazuara në hash, për të garantuar sigurinë afatgjatë në epokën post-kuantike.

III. Si funksionon algoritmi RSA

Algoritmi RSA bazohet në vetitë matematikore të numrave të thjeshtë dhe aritmetikën modulare. Parimi kryesor i RSA përfshin përdorimin e një çifti çelësash: një çelës publik për enkriptim dhe një çelës privat për deshifrim. Çelësat krijohen duke përdorur hapat e mëposhtëm.

1. Gjenerimi i Çelësave:

1) Zgjidh dy numra të mëdhenj të thjeshtë, p dhe q.
2) Llogarit modulin, N, si produkti i p dhe q: $N = p \cdot q$.
3) Llogarit funksionin e Eulerit për N, $\phi(N)$, i cili është numri i numrave të plotë pozitivë më të vegjël se N që janë relativisht të thjeshtë me N. Për dy numra të thjeshtë,
$$\phi(N) = (p-1) \cdot (q-1).$$
4) Zgjidh një numër të plotë e të tillë që $1 < e < \phi(N)$ dhe e është relativisht i thjeshtë me $\phi(N)$. Ky vlerë e do të jetë çelësi publik.
5) Llogarit çelësin privat, d, i cili është inversi modular i e modulo $\phi(N)$. Në mënyrë të tjera, d është vlera që plotëson ekuacionin:
$$(e \cdot d) \equiv 1 \pmod{\phi(N)}.$$
Këtu, $\equiv$ tregon kongruencën modulo.

Çelësi publik i gjeneruar është (N, e), ndërsa ai privat është (N, d). Pasi çelësat janë gjeneruar, mund të ndodhin proceset e enkriptimit dhe dekriptimit:

2. Enkriptimi: Le të supozojmë se mesazhi që do të enkriptohet përfaqësohet si një numër i plotë M, ku $0 \leq M < N$.

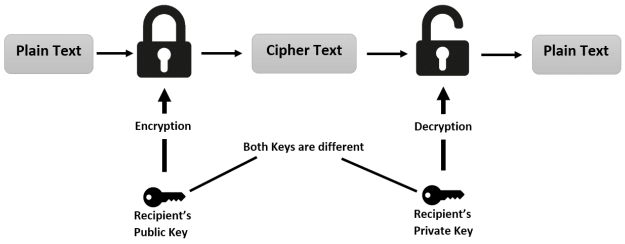

Fig. 1. Algoritmi RSA.

Për të enkriptuar mesazhin M, dërguesi përdor çelësin publik të marrësit (N, e) dhe kryen llogaritjen e mëposhtme:
$$C \equiv M^e \pmod{N}$$
Këtu, C përfaqëson tekstin e shifruar, i cili është forma e koduar e mesazhit M. Operacioni $M^e \pmod{N}$ tregon ngritjen e M në fuqinë e e dhe më pas marrjen e modulit me N. Ky operacion i fuqizimit modular siguron që teksti i koduar të mbetet brenda intervalit prej N.

3. Dekriptimi: Për të deshifruar tekstin e koduar C, marrësi përdor çelësin privat d dhe kryen llogaritjen e mëposhtme:
$$M \equiv C^d \pmod{N}$$
Rezultati M përfaqëson mesazhin origjinal, i cili përftohet duke ngritur tekstin shifror C në fuqinë e d dhe më pas duke marrë modulin e N.

A. Siguria e RSA

Siguria e RSA bazohet në vështirësinë e faktorizimit të numrave të mëdhenj të përbërë. Është e pa realizueshme nga ana e llogaritjeve të përcaktohet eksponenti i çelësit privat d ose faktorët kryesorë p dhe q të modulit N vetëm nga çelësi publik (N, e).

Është e rëndësishme të theksohet se në zbatimet praktike të RSA, mesazhi M zakonisht nuk kodohet drejtpërdrejt si një numër i plotë, por si një seri blloqesh me madhësi fikse. Çdo bllok trajtohet si një numër dhe enkriptohet veçmas, duke përfshirë skema shtesë të mbushjes (padding) për të rritur sigurinë dhe për të parandaluar sulme.

Për më tepër, për të garantuar sigurinë e RSA, gjatësitë e çelësave duhet të zgjidhen siç duhet. Gjatësia e N, e matur në bit, përcakton sigurinë e enkriptimit RSA. Gjatësitë më të gjata të çelësave ofrojnë siguri më të madhe, por kërkojnë më shumë llogaritje për enkriptim dhe dekriptim.

IV. Shembull

Më poshtë paraqitet një shembull i thjeshtë se si ndodh gjenerimi i çelësave, enkriptimi dhe dekriptimi i një mesazhi me anë të algoritmit RSA.

1. Gjenerimi i Çelësave:

1) Zgjedhim dy numra të thjeshtë të vegjël, $p = 11$ dhe $q = 23$.
2) Llogarisim modulin, N, si prodhimin $p \cdot q$:
$$N = p \cdot q = 11 \cdot 23 = 253.$$
3) Llogarisim funksionin e Eulerit të N, $\phi(N)$.
$$\phi(N) = (p-1) \cdot (q-1) = 10 \cdot 22 = 220.$$
4) Zgjedhim një eksponent, e, që është më i vogël se $\phi(N)$ dhe relativisht i thjeshtë me të. Për shembull, le të marrim $e = 31$.
5) Llogarisim eksponentin, d, që është inversi modular i e modulo $\phi(N)$. Ne kërkojmë një vlerë të tillë të d që plotëson ekuacionin
$$(e \cdot d) \equiv 1 \pmod{\phi(N)}.$$
Në rastin tonë, gjejmë me anë të algoritmit të zgjeruar të Euklid-it se $d = 71$.

2. Enkriptimi: Për të enkriptuar një mesazh, përdorim çelësin publik e dhe modulin N. Le të marrim një mesazh të thjeshtë për shembull, $M = 10$. Pasi kemi vlerën e M, mesazhi i enkriptuar C llogaritet si vijon:

$$C \equiv M^e \pmod{N} = 10^{31} \pmod{253} = 43$$

Pra, mesazhi i enkriptuar C është 43.

3. Dekriptimi: Për të dekodifikuar mesazhin C, përdorim çelësin privat d dhe modulin N. Procesi është si vijon:

$$M \equiv C^d \pmod{N} = 43^{71} \pmod{253} = 10$$

Pra, mesazhi origjinal M është 10.

Ky është një shembull i thjeshtë i algoritmit RSA, duke përdorur numra të vegjël për të shpjeguar procesin e enkriptimit dhe dekriptimit. Në praktikë, RSA përdoret me numra shumë më të mëdhenj për të siguruar një nivel të lartë të sigurisë .

V. Konkluzioni

Algoritmi RSA u krijua për të adresuar kufizimet e algoritmeve tradicionale të çelësave simetrik në fushën e kriptografisë. Ai prezantoi konceptin e kriptografisë me çelës publik, duke eliminuar nevojën për kanale të sigurta të shpërndarjes së çelësave. Duke përdorur dy çelësa, RSA lejoi komunikim të sigurt midis palëve që nuk kishin ndërveprim paraprak dhe thjeshtoi menaxhimin e çelësave në rrjete të mëdha. Algoritmi RSA ofroi gjithashtu një nivel më të lartë sigurie duke shfrytëzuar kompleksitetin llogaritës të faktorizimit të numrave të mëdhenj të thjeshtë. Kjo e bëri atë një teknikë të fuqishme dhe të besueshme të kriptimit për të siguruar konfidencialitetin, integritetin dhe autenticitetin e transmetimit të të dhënave.