CARNet
HRVATSKA AKADEMSKA I ISTRAŽIVAČKA MREŽA
CROATIAN ACADEMIC AND RESEARCH NETWORK

Nedostaci PKI infrstrukture

CCERT-PUBDOC-2009-02-255

+CERT.hr u suradnji s LSS

Sigurnosni problemi u računalnim programima i operativnim sustavima područje je na kojem CARNet CERT kontinuirano radi.

Rezultat toga rada je i ovaj dokument, koji je nastao suradnjom CARNet CERT–a i LS&S–a, a za koji se nadamo se da će Vam koristiti u poboljšanju sigurnosti Vašeg sustava.

CARNet CERT, www.cert.hr

Nacionalno središte za **sigurnost računalnih mreža** i sustava.

LS&S, www.LSS.hr

Laboratorij za sustave i signale pri Zavodu za elektroničke sustave i obradbu informacija Fakulteta elektrotehnike i računarstva Sveučilišta u Zagrebu.

Sadržaj

1. Uvod

Kriptiranje podataka ima izvor još u dalekoj povijesti, kada su stari Egipćani prije više od 4000 godina koristili kriptografske sustave za zaštitu informacija. Ipak, pravi razvoj kriptografskih sustava i algoritama donosi pojava i šira uporaba računala (80-tih godina 20. stoljeća). Kriptiranje omogućava prikazivanje podataka pomoću drugih informacija ili u drugom obliku, radi njihove zaštite. Takvi podaci dostupni su samo osobama koje posjeduju određeni ključ za dekriptiranje.

Tijekom povijesti razvijeni su razni algoritmi za kriptiranje podataka, a osnovna podjela je na simetrične i asimetrične. Svaki grupa sadrži više algoritama, od kojih svaki algoritam ima određene prednosti i nedostatke, a primjena određenog algoritma ovisi o zahtjevima korisnika.

PKI (eng. Public Key Infrastructure) infrastrukturu čini skupina programa, sklopovlja, ljudi, sigurnosnih politika i procedura. Sve to je potrebno za kreiranje, upravljanje, pohranjivanje te povlačenje digitalnih certifikata, koji se koriste u kriptografskim sustavima. Digitalni potpisi, zajedno s certifikatima, osiguravaju svojstva poput integriteta, tajnosti, autentičnosti i nemogućnosti poricanja.

Osim navedenih prednosti, PKI infrastruktura donosi i određene sigurnosne probleme. Među najrizičnijima je problem s kolizijom kod MD5 algoritma, korištenog kod većine kriptografskih sustava. Također, postoje određeni rizici prilikom provjere niza certifikata, kao i kod prijenosa parametara javnog ključa. Ovaj dokument osim opisa nedostataka i načina njihove zlouporabe, donosi i osnovne metode zaštite sustava.

2. Kriptiranje

Kriptiranje je postupak transformacije podataka (tzv. *plaintext*) uporabom nekog algoritma u svrhu skrivanja sadržaja od osoba koje ne posjeduju ključ kriptiranja. Rezultat kriptiranja je kriptirana informacija koja se naziva *„ciphertext"*. Pojam kriptiranja podrazumijeva i obrnuti proces kriptiranja tj. dekriptiranje, kako bi se kriptirani podaci mogli pročitati.

Navedeni postupak dugo su koristile organizacije vlade i vojske kako bi osigurali tajnost podataka u komunikaciji. Danas se kriptiranje koristi u zaštiti informacija raznih vrsta civilnih sustava, kao što su računala krajnjih korisnika, mediji za pohranu podataka (npr. USB), mreže (npr. Internet), mobilni telefoni, bežični sustavi, *bluetooth* uređaji i sl. Osim spomenutog, kriptiranje se koristi u upravljanju digitalnim pravima, tj. u sprječavanju neautoriziranog kopiranja ili uporabe digitalnih materijala.

Samo po sebi, kriptiranje može zaštititi tajnost poruke, ali za zaštitu integriteta i autentifikacije (pojmovi objašnjeni u poglavlju 3.1. Osnovne značajke) poruke potrebne su dodatne tehnologije (npr. provjera MAC - *„message authentication code"* vrijednosti ili digitalnih potpisa). Standardi i kriptografski programi koji provode kriptiranje su široko dostupni, ali njihova uspješna uporaba za osiguravanje sigurnosti sustava može biti veliki problem.

Jedna od prvih tehnika kriptiranja je ona koja se temelji na uporabi javnog ključa nazvana PGP (eng. Pretty Good Privacy) kriptiranje. Kreirao ju je Phil Zimmermann 1991.g., a intelektualna prava na algoritam otkupila je tvrtka Network Associates (danas PGP Corporation) 1997.g.

2.1. Svrha i primjena kriptiranja

Kriptiranje se odnosi na algoritam koji informacije prebacuje u oblik koji nije moguće pročitati bez odgovarajućih podataka (ključa). Svrha takvog postupka je osiguravanje privatnosti podataka. Primatelj kriptiranih podataka mora koristiti ključ kako bi dekriptirao tekst poruke u oblik koji je moguće pročitati.

Kriptiranje se može primijeniti na sve pohranjene podatke ili samo na dio podataka. Pri tome, uporaba takvih podataka zahtjeva poznavanje ključa kriptiranja pa napadači ne mogu pročitati osjetljive podatke (jer ne posjeduju ključ). Uređaj za pohranu podataka je tada otporan na napade raznih virusa poput trojanskog konja. Zahvaljujući takvom postupku moguće je spremati financijske podatke ili druge osjetljive informacije na razne medije za pohranu.

Algoritmi za kriptiranje osiguravaju privatnost podataka, ali ne nužno i sigurnost. Za to je potrebno obaviti dodatne mjere, kao što je osiguravanje autentičnosti udaljenog računala (digitalni potpisi i/ili certifikati). U sustavima gdje se koristi neispravan postupak autentifikacije (ili se on uopće ne koristi) napadač može zlouporabiti situaciju za krađu sjednica, pristup osjetljivim informacijama, povećanje prava i sl.

Kriptiranje ima vrlo važnu ulogu u infrastrukturi Internet mreže, jer osigurava sigurnost poruka elektroničke pošte, poruka koje izmjenjuju korisnici, podataka koji se prenose, podataka pohranjenih na poslužiteljima/bazama podataka i sl.

Bez uporabe postupaka kriptiranja, sve informacije proslijeđene preko mreže bile bi dostupne svim korisnicima koji na bilo koji način imaju pristup paketima koji se šalju kroz mrežu. Također, kada se ne bi koristilo kriptiranje podataka prilikom pohrane, sve informacije bi bile dostupne napadaču koji bi ugrozio sigurnost poslužitelja na način da može pristupiti podacima.

Do pojave Internet mreža, kriptiranje je rijetko korišteno u široj javnost, ali je bilo važan alat za vojne potrebe. U današnje vrijeme *online* trgovanja (ponuda/potražnja proizvoda i usluga preko Internet mreže), bankarstva i drugih sličnih usluga potreba za kriptiranjem se uvelike povećala.

2.2. Vrste kriptiranja

Postoje razne vrste kriptiranja, ali nisu sve jednako sigurne i pouzdane. Tehnike kriptiranja se dijele u dvije osnovne kategorije:

1. **Simetrično kriptiranje**:
   - pošiljatelj i primatelj koriste jedan ključ,

- isti ključ koristi se za kriptiranje i dekriptiranje podataka,
- ključ se dogovara prije razmjene podataka,
- primjeri: Blowfish, AES (eng. Advanced Encryption Standard) i DES (eng. Data Encryption Standard).

2. **Asimetrično kriptiranje**:

- kreiranje javnog i privatnog ključa,
- javni ključ je javno dostupan kako bi pošiljatelji mogli kriptirati tekst i slati ga vlasniku tog javnog ključa,
- dekriptiranje se odvija pomoću privatnog ključa koji čini par s javnim ključem,
- viši stupanj sigurnosti (jer je ključ za dekriptiranje u privatnom vlasništvu, a ne izmjenjuje se preko mreže),
- primjeri: RSA i Diffie-Hellman.

U nastavku dokumenta detaljnije su opisane obje vrste kriptiranja podataka.

2.2.1. Simetrično kriptiranje

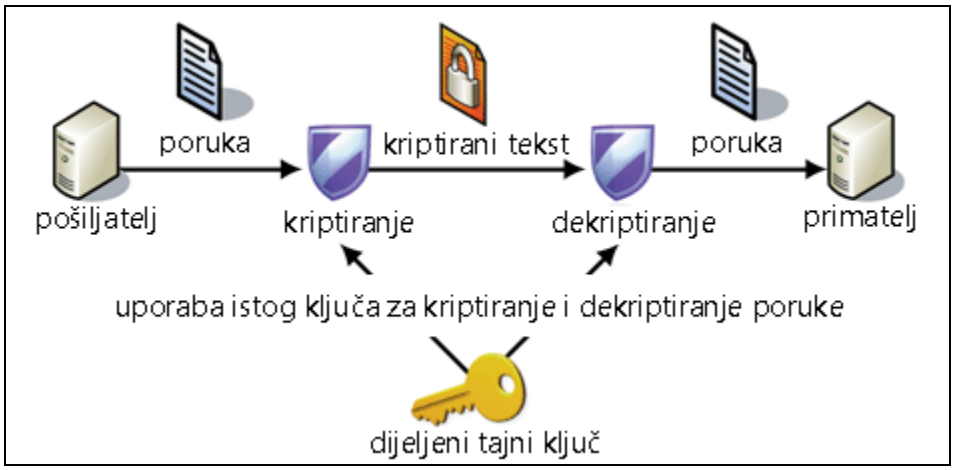
Simetrično kriptiranje ili kriptiranje tajnim ključem je metoda kriptiranja uporabom jedinstvenog ključa, tj. istog ključa za kriptiranje i dekriptiranje podataka. Ključ za dekriptiranje je vezan uz ključ za kriptiranje, na način da ključevi moraju biti identični ili vezani nekom jednostavnom transformacijom (npr. neka vrsta transpozicije), kako bi se podaci mogli uspješno pročitati. U praksi, ovaj tajni ključ predstavlja dijeljene podatke među dva ili više sugovornika u komunikaciji.

Drugi termini za simetrično kriptiranje:

- Kriptiranje tajnim ključem (eng. secret-key),
- Kriptiranje jedinstvenim ključem (eng. single-key),
- Kriptiranje dijeljenim ključem (eng. shared-key),
- Kriptiranje jednim ključem (eng. one-key),
- Kriptiranje privatnik ključem (eng. private-key).

Postupak simetričnog kriptiranja prikazuje slika 1, a provodi se na slijedeći način:

1. pošiljatelj formira poruku,
2. pošiljatelj i primatelj dogovaraju algoritam kriptiranja i tajni ključ,
3. poruka se kriptira uporabom zajedničkog tajnog ključa i dogovorenog algoritma,
4. kriptirani tekst se šalje do primatelja,
5. primatelj dekriptira primljeni tekst koristeći isti tajni ključ i dogovoreni algoritam.



Slika 1. Simetrično kriptiranje

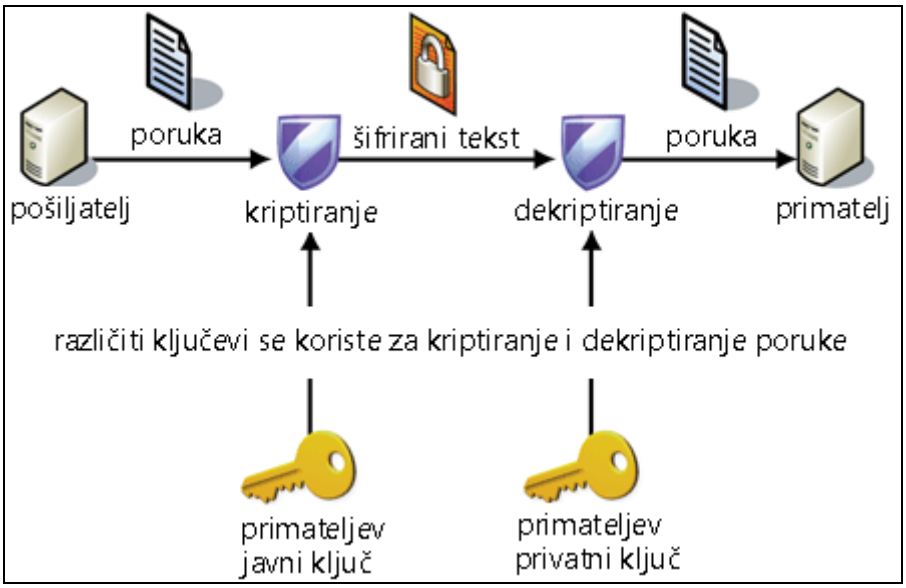
2.2.2. Asimetrično kriptiranje

Asimetrično kriptiranje ili kriptiranje uporabom javnog ključa je metoda za osiguravanje sigurne komunikacije među sugovornicima bez potrebe za prethodnom izmjenom tajnog ključa. Tehnologija je široko rasprostranjena, jer omogućuje siguran prijenos informacija preko Internet mreže.

Metoda koristi različite ključeve za kriptiranje i dekriptiranje podataka: javni i privatni ključ. Privatni ključ je tajan, dok se javni, kao što sami naziv i sugerira, distribuira javnosti. Poruka se kriptira uporabom javnog ključa primatelja, a moguće ju je dekriptirati samo pomoću odgovarajućeg privatnog ključa. Privatni i javni ključ povezani su preko složenih matematičkih relacija, na način da privatni ključ ne može biti izveden iz javnog ključa.

Postupak kriptiranja uporabom javnog ključa prikazuje slika 2, a scenarij kriptiranja poruka je slijedeći:

1. pošiljatelj formira poruku,
2. pošiljatelj kriptira poruku uporabom javnog ključa primatelja i odgovarajućeg algoritma,
3. mrežom se prenosi šifrirani tekst,
4. primatelj dekriptira tekst pomoću svog privatnog ključa i odgovarajućeg algoritma.



Slika 2. Kriptiranje uporabom javnog ključa

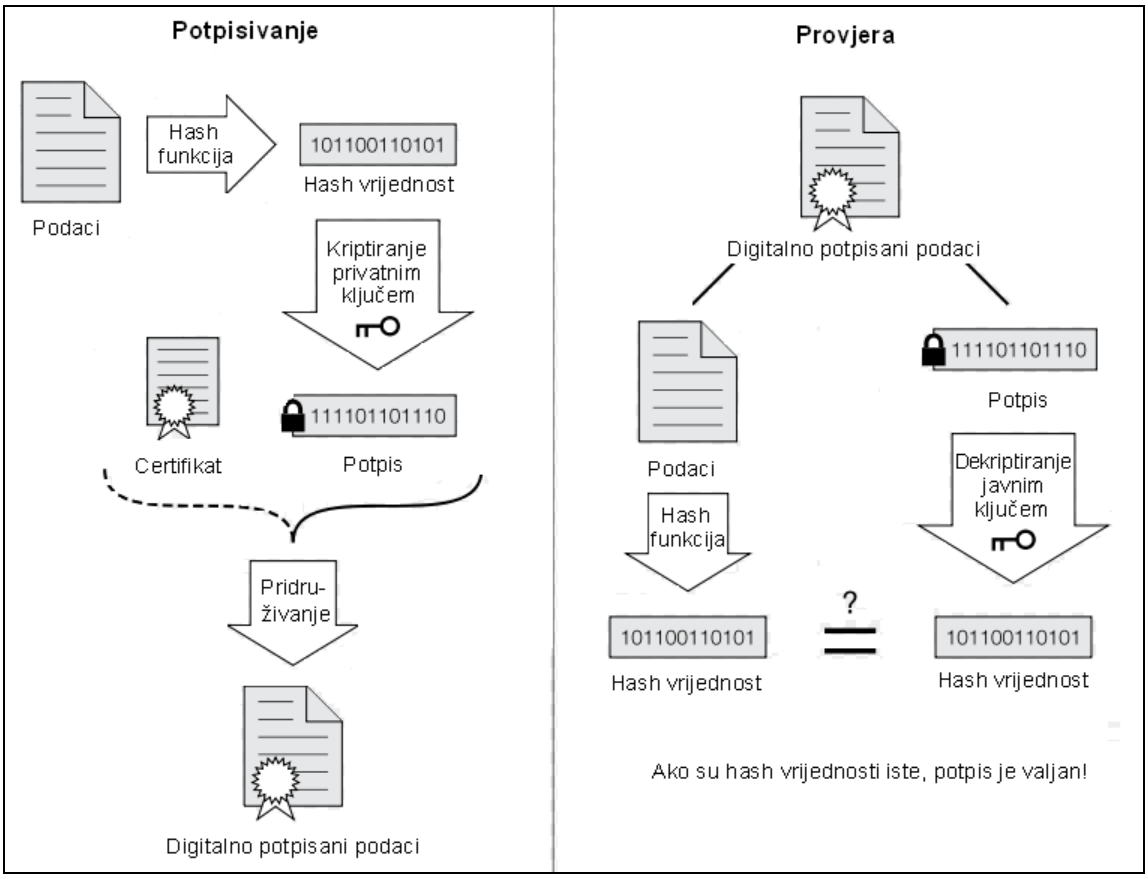
Postoje dva osnovna područja asimetričnog kriptiranja:

Kriptiranje uporabom javnog ključa – poruka kriptirana javnim ključem primatelja ne može se dekriptirati bez privatnog ključa istog primatelja,

Digitalni potpisi – poruku potpisanu pošiljateljevim privatnim ključem može provjeriti svaki korisnik koji ima pristup pošiljateljevom javnom ključu.

Digitalni potpisi služe za potpisivanje podataka te njihovu provjeru (npr. da li su podaci izmijenjeni u prijenosu). Potpisivanje podataka digitalnim potpisom počinje računanjem *hash* vrijednosti podataka pomoću neke *hash* funkcije. Zatim se takva vrijednost kriptira uporabom privatnog ključa korisnika i dobije se digitalni potpis. Kada se potpisu pridruži odgovarajući certifikat, podaci su digitalno potpisani. Provjera takvih podataka vrši se na slijedeći način. Podaci se propuštaju kroz istu *hash* funkciju kako bi se dobila *hash* vrijednost. Digitalni potpis se izdvaja iz digitalno potpisanih podataka te dekriptira uporabom javnog ključa entiteta koji je potpisao podatke. Na taj način se dobije *hash* vrijednost koju je izračunao i umetnuo entitet koji je potpisivao podatke.

Dvije *hash* vrijednosti se uspoređuju i ako su jednake, potpis je valjan. Opisane postupke prikazuje slika 3.



Slika 3. Uporaba digitalnih potpisa

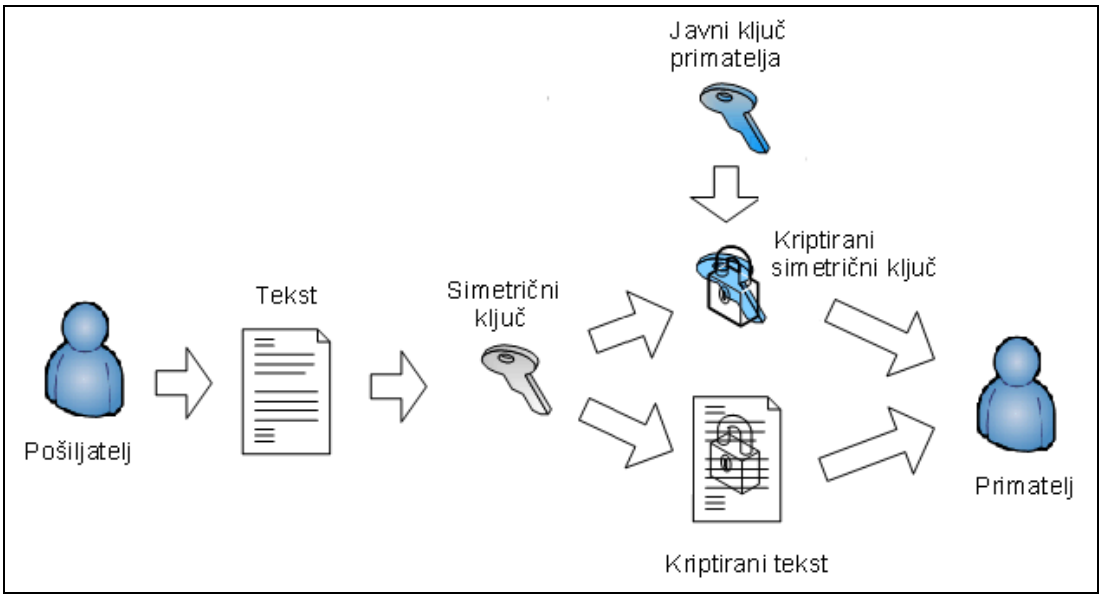
2.2.3. Usporedba simetričnog i asimetričnog kriptiranja

Prednosti i nedostatke opisanih metoda moguće je promatrati kroz slijedeća obilježja:

1. Brzina – simetrični algoritmi su općenito manje računski složeni nego asimetrični algoritmi. U praksi, asimetrični algoritmi su stotine do tisuće puta sporiji od simetričnih.

2. Upravljanje ključevima – nedostatak simetričnih algoritama je zahtjev za dijeljenjem tajnog ključa, tj. postojanje kopije ključa kod svakog sugovornika. Kako bi se osigurala komunikacija ***n*** korisnika potrebno je n(n-1) ključeva i isto toliko kanala za njihov prijenos. Kako bi se smanjio negativan učinak otkrivanja kriptografskih ključeva potrebno ih je redovito mijenjati. Proces odabira, distribucije i pohrane ključeva je poznat pod nazivom upravljanje ključevima.

3. Hibridna kriptografija – u modernim kriptografskim tehnikama, oba opisana postupka koriste se paralelno kako bi se iskoristile prednosti svakoga. Asimetrični algoritmi se koriste za distribuciju simetričnih ključeva na početku sjednice. Kada je jednom simetrični ključ poznat svim korisnicima sjednice, brži simetrični postupci mogu se koristeći taj ključ mogu biti upotrijebljeni za kriptiranje. Navedeni postupak pojednostavljuje problem distribucije ključeva, jer je asimetrični ključ potrebno distribuirati samo u svrhu autentifikacije, a simetrične u svrhu povjerljivosti i autentičnosti. Slika 4 prikazuje opisani postupak hibridne kriptografija.



Slika 4. Hibridna kriptografija

3. PKI

PKI (eng. Public Key Infrastructure) infrastruktura je skupina programa, sklopovlja, ljudi, sigurnosnih politika i procedura potrebnih za kreiranje, upravljanje, pohranjivanje te povlačenje digitalnih certifikata. U kriptografiji, PKI je uređenje koje povezuje javne ključeve s pojedinačnim korisnikom pomoću CA (eng. Certificate Authority) organizacije. Povezivanje je omogućeno kroz procese registracije i izdavanja certifikata koji, ovisno o razini sigurnosti, mogu biti izvedeni programima u CA entitetima ili pod ljudskim nadzorom. Uloga PKI infrastrukture koja osigurava opisano povezivanje naziva se RA (eng. Registration Authority). Za svakog korisnika identitet, javni ključ, povezivanje, uvjeti provjere i drugi atributi zapisuju se u certifikat koji izdaje CA organizacija. Ponekad se za CA organizaciju koristi izraz TTP (eng. trusted third party).

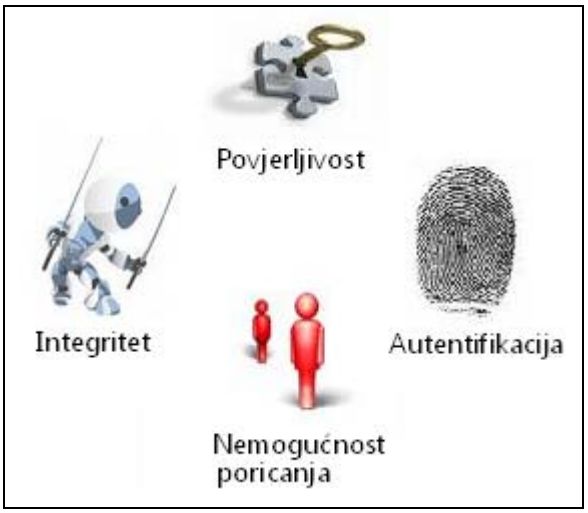
PKI infrastruktura je osnova na kojoj se grade druge aplikacije, sustavi i ostale komponente sigurnosti mreža. Ona čini osnovnu komponentu sigurnosne strategije koja mora surađivati s drugim sigurnosnim mehanizmima, poslovnim praksama i upravljanjem rizicima.

PKI infrastruktura definirana je unutar ITU-T standarda, pod nazivom X.509, a temelji se na strogoj hijerarhijskoj organizaciji izdavanja korisničkih certifikata.

3.1. Osnovne značajke

PKI infrastruktura pruža slijedeća obilježja (slika 5):

1. **Povjerljivost** – osiguravanje tajnosti i privatnosti podataka uporabom kriptografskih algoritama. Osnovni primjeri podataka kojima treba štititi povjerljivost su osobni podaci korisnika, podaci o raznim ugovorima, financijama i sl. Iako je pri tome moguće koristiti i simetričnu i asimetričnu kriptografiju, asimetrična je manje efektivna. Zbog toga se ona koristi za kriptiranje malih dijelova podataka (poput tajnih ključeva, koje koriste simetrični kriptografski sustavi). Simetrični postupci su obično ugrađeni u PKI infrastrukturu za potrebe kriptiranja velikih količina podataka.
2. **Integritet** – osiguravanje da podaci ne mogu biti ugroženi ili izmijenjeni, a prijenos nepromijenjen. Osnovni primjeri podataka kojima se treba zaštititi integritet u prijenosu su javni ključevi, certifikati i digitalni potpisi. Često u takve podatke pripadaju i sadržaj poruka, elektronička pošta, ugovori te ostale važne informacije. Integritet može biti osiguran unutar PKI infrastrukture uporabom simetrične i asimetrične kriptografije. Primjer uporabe simetrične kriptografije za očuvanje integriteta je DES algoritam pri generiranju MAC (eng. Message Authentication Code) vrijednosti. Asimetrične metode se obično koriste zajedno sa SHA-1 ili MD5 algoritmima za osiguravanje integriteta. Dobro dizajnirana PKI infrastruktura mora koristiti protokole koji zahtijevaju uporabu navedenih algoritama kako bi osigurale efektivne mehanizme za zaštitu integriteta.
3. **Autentifikacija** – provjera identiteta entiteta uporabom certifikata i digitalnih potpisa. Primjer korištenja autentifikacije su usluge stvaranja i prodaje proizvoda preko elektroničkih sustava („*e-commerce*" usluge), gdje se primjenjuju kriptografski sustavi temeljeni na javnim ključevima. Osnovna prednost autentifikacije u PKI infrastrukturi je podrška udaljenim korisnicima, a postupak se zasniva na matematičkoj ovisnosti između javnog i privatnog ključa. Poruku koju potpiše jedan entitet može provjeriti bilo koji drugi pouzdani entitet.
4. **Nemogućnost poricanja** – osiguravanje nemogućnosti poricanja i odbijanja prijenosa podataka uporabom kriptografije javnim ključevima i digitalnih potpisa. Primjer usluga u kojima se zahtjeva navedeno svojstvo su *e-commerce* usluge (razmjena informacija, ugovora i sl.). Kada se podatak kriptira uporabom privatnog ključa, svaki korisnik koji ima pristup odgovarajućem javnom ključu može odrediti da je samo vlasnik privatnog ključa mogao potpisati poruku. Zbog opisanog postupka nužno je da krajnji sustavi osiguravaju sigurnost privatnih ključeva korištenih u digitalnom potpisivanju podataka.



Slika 5. Obilježja PKI infrastrukture

3.2. Uporaba i prednosti

Osnovna funkcija PKI infrastrukture je omogućavanje distribucije i korištenja javnih ključeva i certifikata osiguravajući sigurnost i integritet. Ona čini osnovu na kojoj se grade druge aplikacije i ostale sigurnosne komponente mreža.

Sustavi koji često zahtijevaju sigurnosne mehanizme temeljene na PKI infrastrukturi uključuju:

- poruke elektroničke pošte,
- razne aplikacije (npr. aplikacije za ankete, identifikaciju i sl.),
- usluge stvaranja i prodaje proizvoda preko elektroničkih sustava (*„e-commerce"* usluge),
- ponudu financijskih usluga preko Internet mreže (*„online"* bankarstvo) i
- razmjenu poruka preko mreže.

PKI omogućava slijedeće osnovne sigurnosne usluge:

- SSL, IPsec i HTTP protokole za komunikaciju i sigurnost,
- S/MIMIE i PGP protokole za sigurnost razmjene poruka elektroničke pošte,
- SET protokol za razmjenu vrijednosti (eng. value exchange) i
- podršku za B2B (eng. Business-to-business) poslovanje.

Osnovne prednosti uporabe PKI infrastrukture:

- smanjenje troškova obrade transakcijskih pristupa,
- smanjenje i odjeljivanje rizika,
- povećanje efektivnosti i performansa sustava i mreža te
- smanjenje kompleksnosti sigurnosnih sustava.

Razna rješenja zasnivaju se na prednostima uporabe javnih ključeva i kriptografije:

- identifikacijiski brojevi (ID) za studente na sveučilištima,
- glasovanje,
- anonimna razmjena informacija,
- identifikacija,
- *online* karte za putovanja,
- distribucija programa,
- razne vrste provjera podataka i
- upravljanje ključevima i sl.

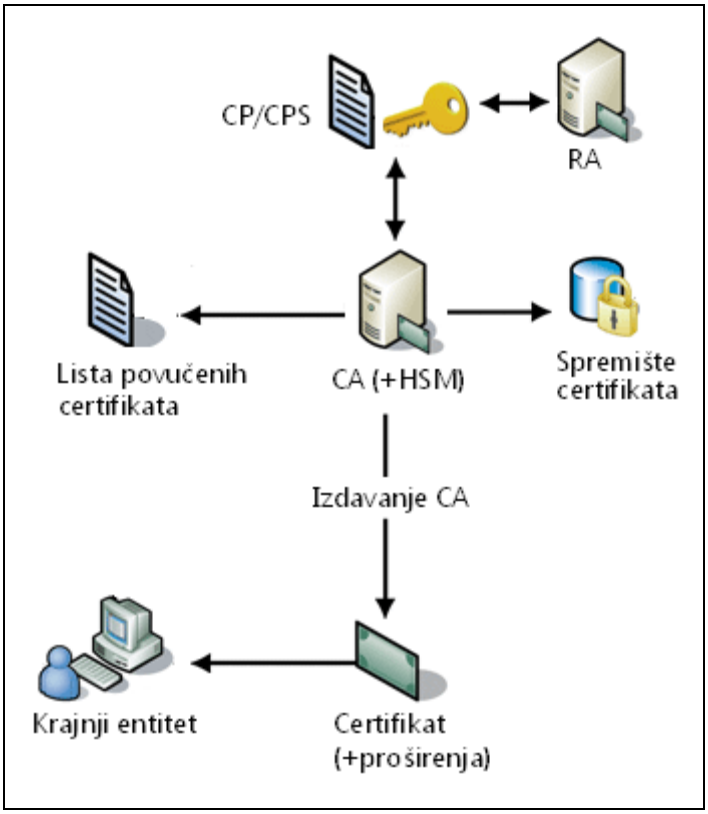
3.3. Arhitektura infrastrukture

PKI okruženje sadrži sigurnosne politike, usluge i protokole koji služe kao podrška kriptografiji javnih ključeva za upravljanje ključevima i certifikatima. Generiranje, distribucija i upravljanje ključevima i certifikatima obično vrše CA (eng. Certification Authorities), RA (eng. Registration Authorities) te druge usluge (eng. directory services). One se koriste kako bi uspostavile hijerarhiju ili niz povjerenja (eng. chain of trust) te omogućavaju uporabu digitalnih certifikata za identifikaciju raznih entiteta.

Prilikom komunikacije preko Internet mreža, entiteti koji nisu poznati jedan drugome, svaki zasebno uspostavi vezu sa CA entitetom. Nakon obavljanja autentifikacije, prema pravilima u CPS (eng. Certificate Practices Statement) zapisu, CA pruža svakome entitetu digitalni certifikat. Dodijeljene certifikate potpisuje CA entitet što je jamstvo svakome entitetu u komunikaciji pa oni mogu uspostaviti vezu.

3.3.1. Komponente sustava

Infrastrukturu se sastoji od više međusobno povezanih objekata, aplikacija i usluga kako je prikazano na slici 6.



Slika 6. Komponente PKI infrastrukture

Osnovne komponente sustava su:

1. **Krajnji entiteti** ili entiteti za potpisivanje – svaki korisnik ili objekt (poput računala) koji trebaju digitalni certifikat iz nekog razloga. Moraju imati sposobnost generiranja javnog ključa te pohrane i korištenja privatnog ključa.
2. **CA** (eng. Certificate Authorities) – entiteti kojima „vjeruje“ jedan ili više korisnika, a služe za kreiranje i dodjelu javnih ključeva certifikata. Ima ulogu treće povjerljive strane i pruža razne usluge upravljanja ključevima. Razina povjerenja CA ovisi o razini suglasnosti drugih entiteta u tom CA. Razina suglasnosti ovisi o politikama i procedurama koje je CA uspostavila za određeni korisnički identitet. CA javni ključ se dostavlja svim entitetima koji vjeruju tom CA. Osnovne funkcije CA su: generiranje i povlačenje certifikata. CA radi unutar konteksta

poslovne politike poznate kao CP (eng. Certificate Policy) te funkcionira prema CPS (eng. Certificate Practices Statement) zapisima.

3. **CP** (eng. Certificate Policy) – skup pravila koja uključuju primjenjivost javnih ključeva certifikata za određenu zajednicu ili klasu aplikacija s osnovnim sigurnosnim zahtjevima (npr. CP može ukazivati na primjenjivost nekog tipa certifikata dobivenog uporabom javnog ključa na osiguravanje autentičnost izmjene elektroničkih podataka za trgovanje dobrima određene novčane vrijednosti).
4. **CPS** (eng. Certificate Practices Statement) – prakse koje CA uključuje u izdavanje ključeva. Treba definirati sve procese uključene u generiranje, izdavanje, upravljanje, pohranu, dostavljanje i povlačenje javnih ključeva. Svaka PKI implementacija treba imati slijedeće CPS:
    a. svrha PKI,
    b. posebni poslovni zahtjevi,
    c. sigurnosna arhitektura,
    d. odgovarajući model povjerenja i
    e. posebne sigurnosne usluge koje podržavaju PKI.

    Osim toga, CPS entiteti definiraju procese autentifikacije za svaki entitet.
5. **HSM** (eng. Hardware Security Modules) – osnovna komponenta CA entiteta, koja omogućava uspostavu povjerenja ne samo s klijentima određenog CA, nego i svima koji ovise o certifikatima izdanih od krajnjih entiteta. Budući da povjerenje ovisi o sigurnosti i integritetu privatnih ključeva korištenih za potpisivanje javnih ključeva certifikata, neophodno je osigurati najveću razinu zaštite tih ključeva. CA pohranjuje i koristi privatne ključeve određene u HSM modulima, zvanim još i TRSM (eng. Tamper Resistant Security Modules) moduli.
6. **Javni ključ certifikata** (eng. Public Key Certificates) – služi kao potvrda povezivanja identiteta krajnjeg korisnika i njegovog javnog ključa. Sadrži dovoljno informacija kako bi drugi entitet mogao provjeriti i potvrditi identitet vlasnika certifikata. Osnovna građa certifikata uključuje:
    a. Ime entiteta,
    b. Informacije za identifikaciju entiteta,
    c. vrijeme trajanja valjanosti certifikata,
    d. javni ključ krajnjeg entiteta.

    Obično se koristi format certifikata opisan u IETF X.509 standardu (opis dostupan u IETF RFC 2459).
7. **Proširenja certifikata** (eng. Certificate Extensions) – pružaju dodatne informacije o certifikatu i dopuštaju njihovu uporabu za posebne potrebe organizacije. Entiteti moraju poznavati proširenja certifikata kako ne bi došlo do negativnog utjecaja. Informacije koje se nalaze u proširenju certifikata su:
    a. Politika,
    b. Uporaba,
    c. Povlačenje.
8. **RA** (eng. Registration Authorities) – provodi određene administrativne i sliče zadatke u korist CA. Osnovna uloga RA je provjera identiteta krajnjeg entiteta i određivanje ovlasti za dodjelu javnog ključa. RA mora provesti sve politike i procedure definirane u CP i CPS kako bi ispitala zahtjev za certifikatom. Pri tome provjerava ime, podatke za provjeru, javni ključ, proširenja certifikata i ostale informacije.
9. **Spremište certifikata** (eng. Certificate Depositories) - služi za distribuciju certifikata na način da je svaki objavljeni certifikat spremljen u spremište koje kontrolira CA i RA. Tada je proces distribucije pojednostavljen, jer je prilikom izdavanja novog certifikata potrebno samo obnoviti zapise u spremištu.
10. **Lista povučenih certifikata** (eng. Certificate Revocation List) – popis svih povučenih certifikata, kojeg obnavljaju RA i CA.

3.3.2. Modeli

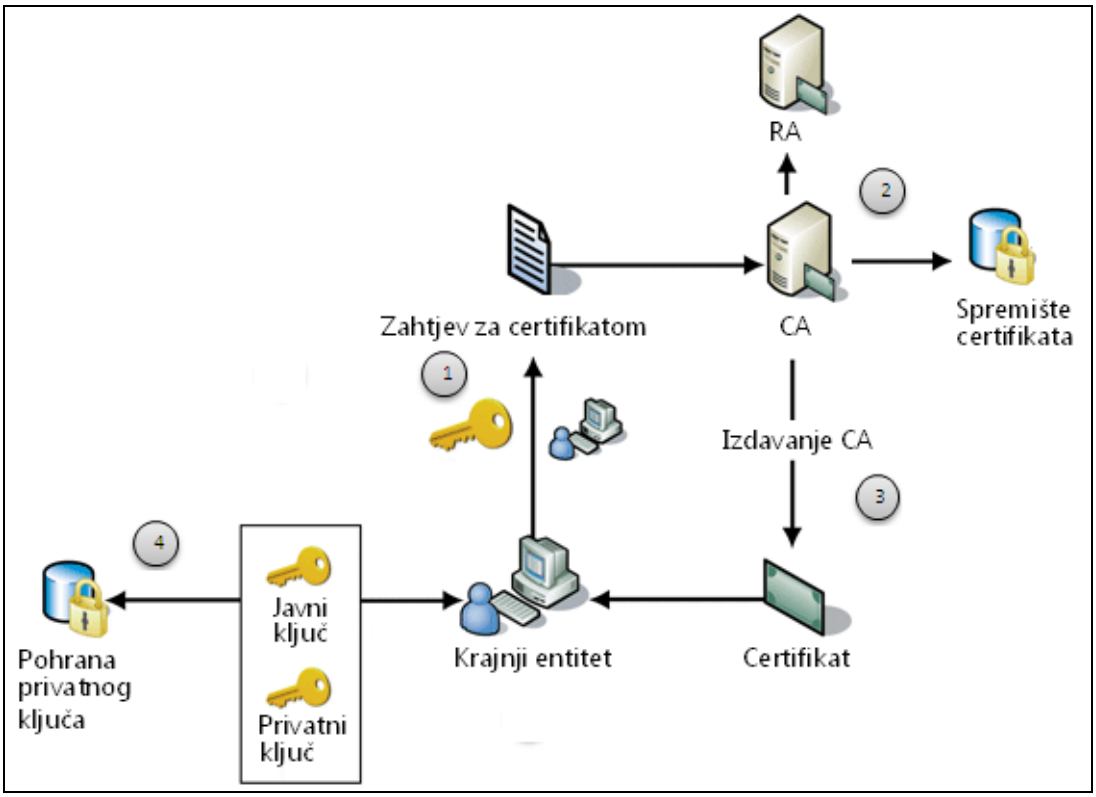
Postoje tri osnovna modela:

1. **Hijerarhijski model**:
   - tipična implementacija PKI infrastrukture,
   - dozvoljava CA organizaciji potpisivanje certifikata krajnjih entiteta,
   - hijerarhija sadrži skupinu CA entiteta koji su organizirani na temelju prethodno utvrđenih pravila,
   - postoji točka povjerenja (eng. trust point) za svaki stvoreni certifikat,
   - prednost: efektivna, proširiva i podesiva PKI infrastruktura,
   - nedostatak: puno CA entiteta znači i više mogućnosti za sigurnosne rizike (rušenje jednog entiteta ugrožava cijelu infrastrukturu).
2. **Distribuirani model**:
   - ne uključuje CA niti neki drugi oblik organizacije za provjeru identiteta korisnika,
   - uporaba kod PGP (eng. Pretty Good Privacy) postupka za poruke elektroničke pošte,
   - nedostatak: krajnji entiteti sami određuju stupanj povjerenja koji dodjeljuju ostalim entitetima.
3. **Izravni model**
   - nazvan i model od točke do točke (eng. Peer to Peer),
   - koristi se unutar postupaka temeljenih na kriptografiji tajnog ili simetričnog ključa,
   - ne postoji treća povjerljiva strana (CA i sl.),
   - entiteti uspostavljaju vezu od točke do točke te sami određuju razinu povjerenja.

3.4. Funkcionalnost

Osnovne funkcije PKI infrastrukture su:

1. **kriptografija uporabom javnog ključa**:
   - uključuje generiranje, distribuciju, administraciju i kontrolu kriptografskih ključeva,
   - moguća uporaba „Diffie i Hellman" te RSA (eng. Rivest, Shamir i Adleman) algoritma,
   - algoritmi se zasnivaju na složenim matematičkim relacijama,
   - korištenje javnog i privatnog ključa.
2. **izdavanje certifikata** (slika 7):
   - povezivanje javnog ključa jednog korisnika, organizacije ili drugih entiteta sa samim entitetom,
   - entitet može poslati zahtjev za certifikatom,
   - zahtjev provjeravaju CA i RA te vrše provjeru identiteta,
   - CA formira certifikat na temelju zahtjeva,
   - CA potpisuje certifikat s privatnim ključem te zapisuje u spremište certifikata,
   - krajnji entitet sprema privatni ključ, a javni distribuira.
3. **provjera certifikata**:
   - provjera da certifikat postoji i da je valjan,
   - poruke potpisane valjanim certifikatima zadovoljavaju svojstva povjerljivosti, integriteta, autentičnosti te nemogućnosti poricanja,
   - svi valjani certifikati spremljeni su u spremište certifikata.
4. **povlačenje certifikata**:
   - povlačenje prethodno izdanih certifikata,
   - objava odluke na CRL popisu.



Slika 7. Izdavanje certifikata

3.5. CA certifikati

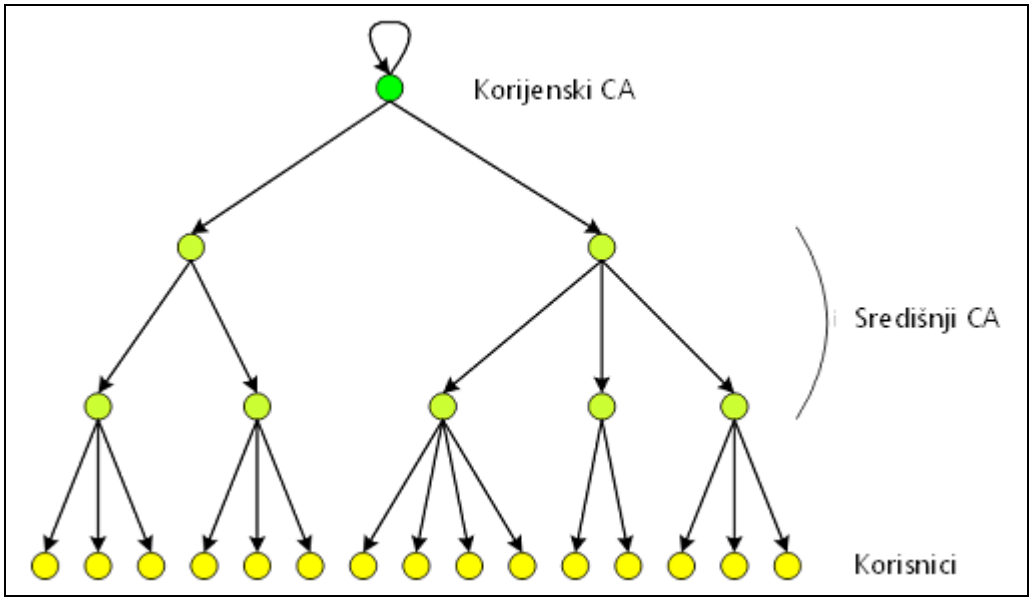
Certifikati su sredstva koja se koriste kako bi osigurala pouzdan prijenos osjetljivih informacija preko mreže. To su dokumenti koji sadrži identitet i javni ključ, povezane digitalnim potpisom koji kreira CA. X.509 certifikat (slika 8) sadrži slijedeće dijelove:

1. **dio koji je potrebno „potpisati“**:
   a. serijski broj,
   b. vrijeme valjanosti,
   c. ime izdavatelja (identitet CA),
   d. ime entiteta (krajnji korisnik, drugi CA i sl.),
   e. javni ključ,
   f. osnovna polja s zahtjevima:
      - oznaka da li se radi o CA ili korisničkom certifikatu,
      - duljina puta – maksimalni dozvoljeni broj CA entiteta između korijenskog CA i korisnika,
2. **dio s potpisom**:
      - potpis preko svih dijelova koje treba „potpisati“,
      - koristi se za generiranje privatnog ključa.

```
Certificate:
   Data:
       Version: 1 (0x0)
       Serial Number: 7829 (0x1e95)
       Signature Algorithm: md5WithRSAEncryption
       Issuer: C=ZA, ST=Western Cape, L=Cape Town, O=Thawte Consulting cc,
               OU=Certification Services Division,
               CN=Thawte Server CA/emailAddress=server-certs@thawte.com
       Validity
           Not Before: Jul  9 16:04:02 1998 GMT
           Not After : Jul  9 16:04:02 1999 GMT
       Subject: C=US, ST=Maryland, L=Pasadena, O=Brent Baccala,
                OU=FreeSoft, CN=www.freesoft.org/emailAddress=baccala@freesoft.org
       Subject Public Key Info:
           Public Key Algorithm: rsaEncryption
           RSA Public Key: (1024 bit)
               Modulus (1024 bit):
                   00:b4:31:98:0a:c4:bc:62:c1:88:aa:dc:b0:c8:bb:
                   33:35:19:d5:0c:64:b9:3d:41:b2:96:fc:f3:31:e1:
                   66:36:d0:8e:56:12:44:ba:75:eb:e8:1c:9c:5b:66:
                   70:33:52:14:c9:ec:4f:91:51:70:39:de:53:85:17:
                   16:94:6e:ee:f4:d5:6f:d5:ca:b3:47:5e:1b:0c:7b:
                   c5:cc:2b:6b:c1:90:c3:16:31:0d:bf:7a:c7:47:77:
                   8f:a0:21:c7:4c:d0:16:65:00:c1:0f:d7:b8:80:e3:
                   d2:75:6b:c1:ea:9e:5c:5c:ea:7d:c1:a1:10:bc:b8:
                   e8:35:1c:9e:27:52:7e:41:8f
               Exponent: 65537 (0x10001)
   Signature Algorithm: md5WithRSAEncryption
       93:5f:8f:5f:c5:af:bf:0a:ab:a5:6d:fb:24:5f:b6:59:5d:9d:
       92:2e:4a:1b:8b:ac:7d:99:17:5d:cd:19:f6:ad:ef:63:2f:92:
       ab:2f:4b:cf:0a:13:90:ee:2c:0e:43:03:be:f6:ea:8e:9c:67:
       d0:a2:40:03:f7:ef:6a:15:09:79:a9:46:ed:b7:16:1b:41:72:
       0d:19:aa:ad:dd:9a:df:ab:97:50:65:f5:5e:85:a6:ef:19:d1:
       5a:de:9d:ea:63:cd:cb:cc:6d:5d:01:85:b5:6d:c8:f3:d9:f7:
       8f:0e:fc:ba:1f:34:e9:96:6e:6c:cf:f2:ef:9b:bf:de:b5:22:
       68:9f
```

Slika 8. Primjer X.509 certifikata

Certifikati zadovoljavaju hijerarhijsku strukturu. Potpis podataka može se provjeriti pomoću javnog ključa entiteta koji potpisuje podatke. Taj javni ključ je povezan s korisnikovim identitetom preko certifikata, što je moguće provjeriti preko potpisa certifikata, koristeći javni ključ CA koji izdaje određeni certifikat. Javni ključ CA entiteta nalazi se unutar CA certifikata, koji je sloj više u hijerarhiji. Također, CA certifikat tog CA entiteta potpisuje CA na još jednom višem sloju u hijerarhiji. Na vrhu hijerarhije nalazi se korijenski CA entitet, koji „potpisuje sam sebe“. Opisanu hijerarhiju prikazuje slika 9.



Slika 9. Hijerarhijska struktura CA entiteta

4. Nedostaci PKI infrastrukture

4.1. MD5 problem

30. prosinca 2008. godine tim stručnjaka, na kongresu " Chaos Communication Congress" u Berlinu, objavio je da su uspjeli kreirati lažni certifikat te ga prikazati valjanim. Tim se sastojao od slijedećih članova: Alexander Sotirov, Marc Stevens, Jacob Appelbaum, Arjen Lenstra, David Molnar, Dag Arne Osvik i Benne de Weger.

U nastavku su opisna prijašnja otkrića nepravilnosti kod MD5 algoritma, a novi postupak zlouporabe nalazi se u poglavlju „Iskorištavanje MD5 problema".

4.1.1. MD5 algoritam

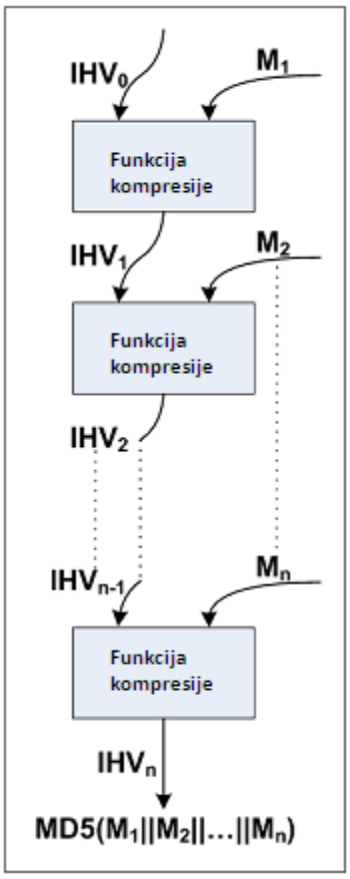
MD5 algoritam koristi Merkle-Damgård iterativnu konstrukciju. Ulazni niz bitova se dopuni do višekratnika od 512 okteta te dijeli u ulazne blokove (blokovi veličine 512 okteta).

Jezgra MD5 algoritma je funkcija kompresije. MD5 algoritam učitava ulazne blokove u uzastopnom pozivanju funkcije kompresije, koja koristi svaki blok smanjen na 128 okteta. Takvi blokovi se nazivaju IHV (eng. Intermediate Hash Value). Stoga, funkcija kompresije koristi dva ulaza: 128 okteta IHV niza i 512 okteta ulaznog bloka. Inicijalno stanje naziva se fiksna vrijednost, a izlaz funkcije *hash* vrijednost.

Znači, ako dopunimo ulazne blokove s M_1, M_2, ..., M_n (do 512 okteta svaki), za inicijalni IHV s IHV_0 (128 okteta) i funkciju kompresije CF, sekvenca kompresije je:

$$
\begin{aligned}
&IHV_0 = \text{fiksna vrijednost}\\
&IHV_1 = CF(IHV_0, M_1)\\
&IHV_2 = CF(IHV_1, M_2)\\
&IHV_3 = CF(IHV_2, M_3)\\
&\quad\vdots\\
&\quad\vdots\\
&\quad\vdots\\
&IHV_{n-1} = CF(IHV_{n-2}, M_{n-1})\\
&IHV_n = CF(IHV_{n-1}, M_n)\\
&\text{MD5 hash} = IHV_n
\end{aligned}
$$

Opisani postupak uporabe funkcije kompresije prikazan je na slici 10.



Slika 10. MD5 algoritam

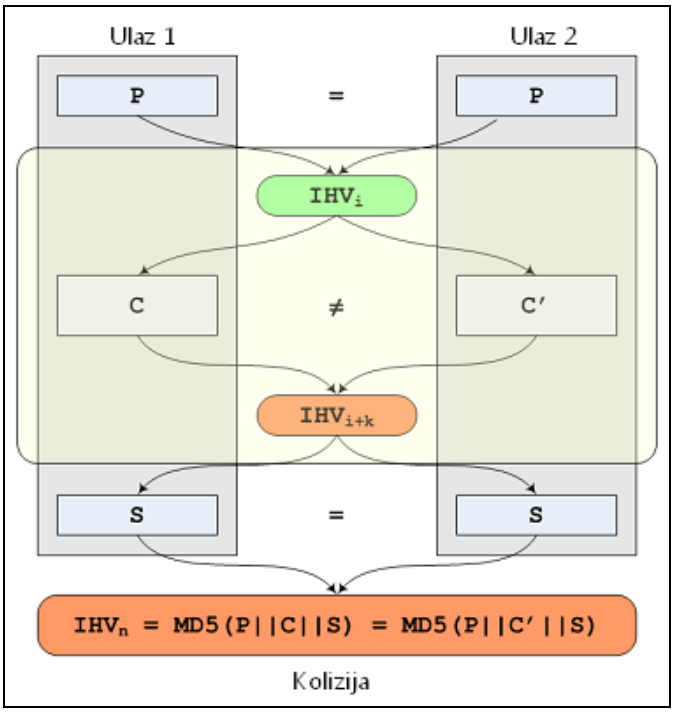
4.1.2. MD5 kolizija

Njihov rad se zasniva na ranijem otkriću stručnjaka Xiaoyun Wang i Hongbo Yu, koji su 2004. g. otkrili MD5 koliziju (slika 11). Drugim riječima, njihova metoda stvara ulazne vrijednosti kao bilo koji 128 oktet IHV_0 parova {{M_1,M_2}, {M_1',M_2'}}, a svaki sadrži 2 ulazna bloka od 512 okteta takva da vrijedi:

$$\{M_1,M_2\} \neq \{M_1',M_2'\}$$
$$IHV_0 = IHV_0'$$
$$IHV_1 = CF(IHV_0,M_1) \neq IHV_1' = CF(IHV_0',M_1')$$
$$IHV_2 = CF(IHV_1,M_2) = IHV_2' = CF(IHV_1',M_2')$$

Zahvaljujući iterativnoj strukturi MD5 algoritma i činjenici da IHV_0 može imati bilo koju vrijednost od 128 okteta, takva kolizija može biti kombinirana u veće ulazne vrijednosti. Za svaki prefiks P i dani sufiks S, par blokova {C,C'} može biti izračunat tako da vrijedi:

$$MD5(P||C||S) = MD5(P||C'||S)$$



Slika 11. MD5 kolizija

Osnovna ideja bila je skriti slučajne kolizijske blokove u javni ključ. To ne bi izazvalo sumnju ni pri pažljivom pregledu certifikata u svrhu pronalaska neispravnosti. Kasnije je otkriveno da je moguće skriti takve blokove u RSA module. Tako je prikazano kako je moguće proizvesti različite X.509 certifikate s identičnim MD5 sumama (eng, hash) potpisanih dijelova. To znači da takvi certifikati imaju iste potpise, što narušava osnovne principe infrastrukture. Ipak, budući da prefiksi ulaza moraju biti isti kako bi certifikati bili prikazani kao isti, zlouporaba ima određena ograničenja.

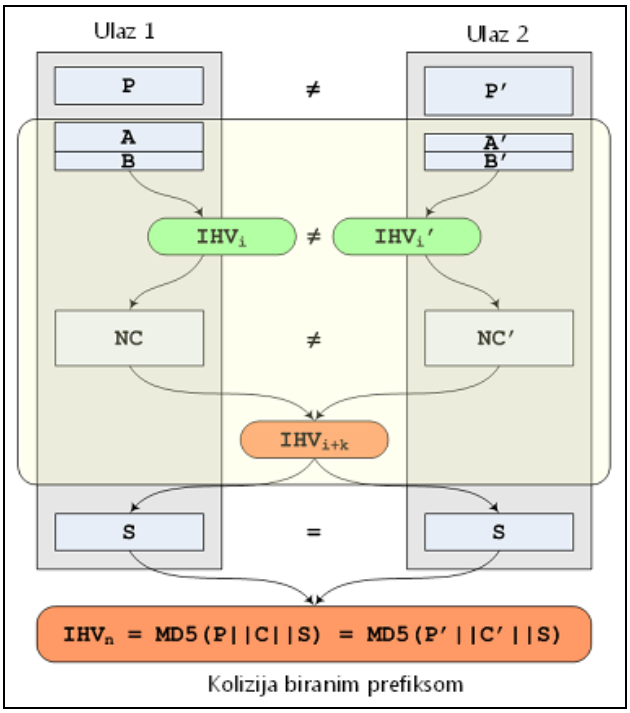
4.1.3. Kolizija uporabom posebno odabranih prefiksa

2007. g. Marc Stevens proširio je opisani postupak kako bi dobio metodu poznatu pod nazivom kolizija uporabom posebno odabranih prefiksa (eng. chosen-prefix collisions). Ova nova metoda pokazuje da se kolizija može izazvati i između dvije različite početne IHV vrijednosti, ali zahtjeva više od dva ulazna bloka.

I ovaj postupak iskorištava iterativnu strukturu MD5 algoritma, a može za svaki ulazni par prefiksa {P,P'} i svaki sufiks S proizvesti blok {C,C'}, takav da vrijedi:

MD5(P||C||S) = MD5(P'||C'||S).

Slika 12 prikazuje spomenutu koliziju, a par {C,C'} se konstruira na slijedeći način. Prvo su P i P' povećani s nizovima A i A' na duljinu da se postigne ista duljina P||A i P'||A'. Koristeći tzv. "birthdaying" korak stvaraju se B i B' na način da P||A||B i P'||A'||B' imaju istu duljinu (višestruki blokovi od 512 okteta). Krajnje IHV vrijednosti imaju prethodno određenu strukturu. To omogućuje stvaranje {NC,NC'} para (eng. near collision blocks), koji sadrže više ulaznih blokova veličine 512 okteta. Ako vrijedi: C = A||B||NC i C' = A'||B'||NC', krajnje IHV vrijednosti su identične. Prema tome, MD5(P||C) = MD5(P'||C') pa je također za svaki sufiks vrijedi MD5(P||C||S) = MD5(P'||C'||S). Opisani postupak je izvediv u praksi, ali dosta je složeniji od prethodnog postupka.



Slika 12. Kolizija uporabom posebno odabranih prefiksa

Ova metoda pokazala je kako je moguće izabrati bilo koji par prefiksa te dobiti različite certifikate s istim MD5 *hash* vrijednostima potpisanih dijelova, tj. identičnim potpisima.

Napadač mora predvidjeti vrijednosti određenih polja (vrijeme valjanosti, serijski broj i sl.) koja inače ne kontroliraju korisnici. Uspješno predviđanje takvih vrijednosti zahtjeva kontrolu procedura CA entiteta.

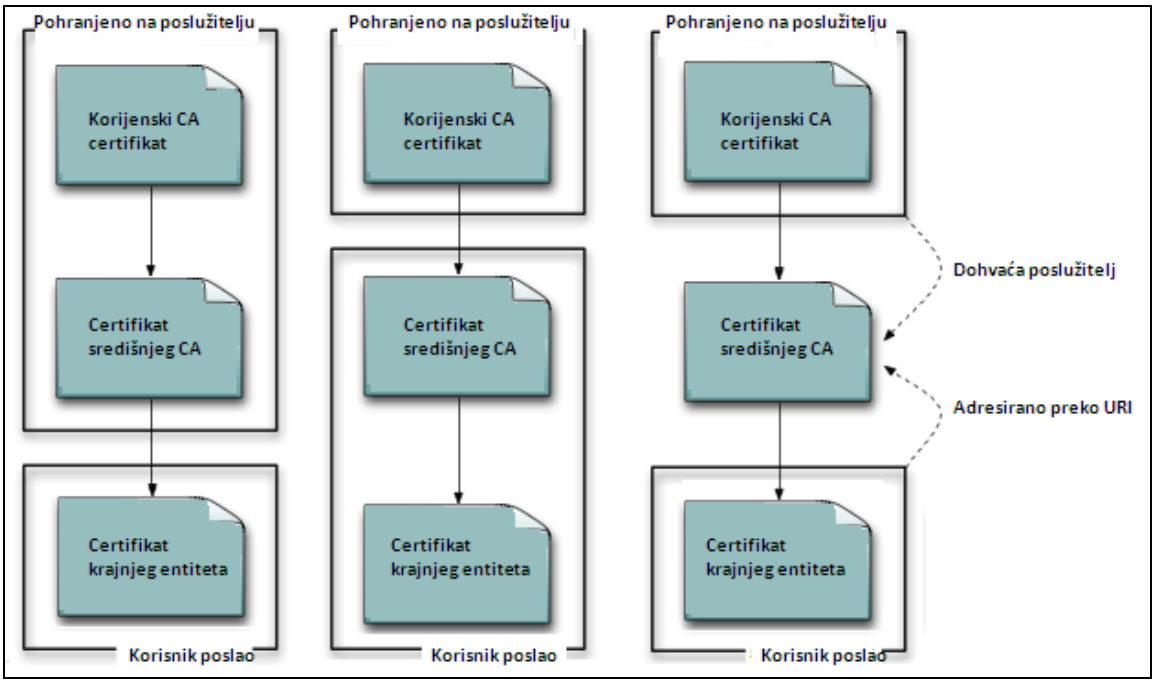
4.2. HTTP i PKI

Još jedan problem kod PKI infrastrukture javlja se zbog neodgovarajuće implementacije sustava, na način preporučen u RFC 3280 - Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile dokumentu. Neautentificiranom korisniku propust omogućava slanje proizvoljnog HTTP zahtjeva s računala koje obrađuje poslani certifikat.

4.2.1. Opis problema

Na početku je prikazan rad s nizom certifikata, kako bi se bolje shvatio spomenuti nedostatak. U tipičnom sustavu s jednim korijenskim CA entitetom i jednim središnjim CA entitetom, postoje tri osnovna načina rukovanja provjerom puta certifikata.

Slika 13 prikazuje različite načine na koje središnji CA entitet može biti uključen u provjeru certifikata. Prvi način uključuje pohranu certifikata, koji pripadaju CA entitetu, na poslužitelj. Potreba pohranjivanja certifikata na poslužitelj je ujedno i osnovni nedostatak ove metode. Drugi način uključuje postupak u kojem korisnik šalje certifikat središnjem CA, a osnovni nedostatak je potreba prilagođavanja korisničkih računala. Zadnji način je korištenje posebnih informacija o pristupu kako bi se specificirale URI (eng. Uniform Resource Identifier) adrese središnjih CA entiteta unutar certifikata krajnjih korisnika. Poslužitelj zatim dohvaća certifikat središnjeg CA entiteta preko dane URI adrese i koristi ga za provjeru niza certifikata. Također, moguće je privremeno pohraniti dohvaćeni certifikat.

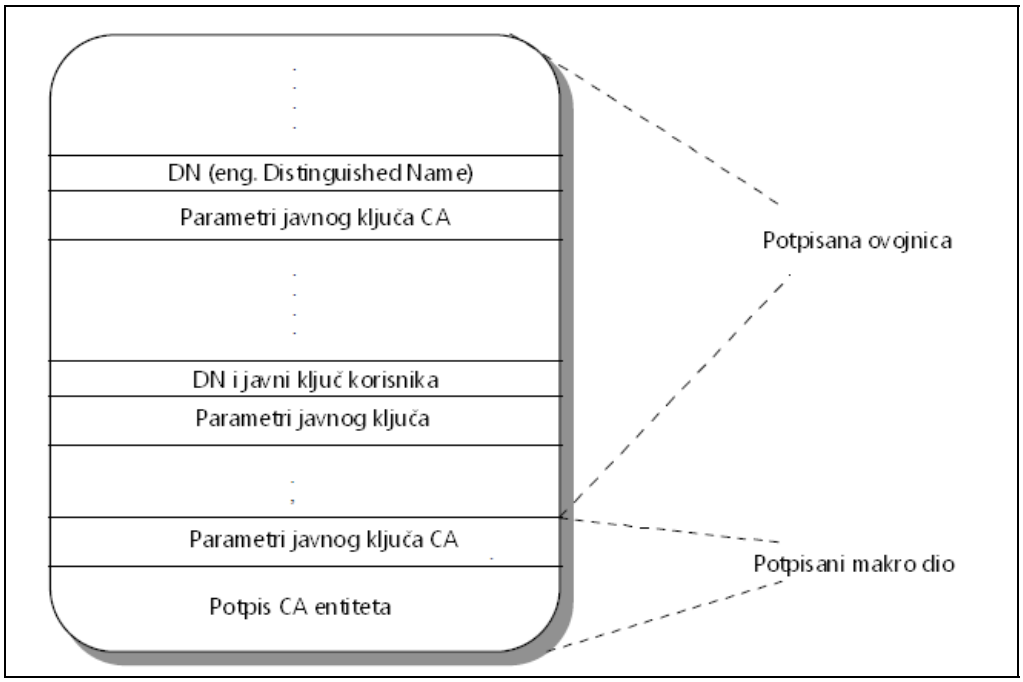


Slika 13. Metode provjere niza certifikata

Problem se javlja kod zadnjeg opisanog načina provjere certifikata. U tom slučaju, dok se ne provjeri niz certifikata, korisniku koji ih šalje se „ne vjeruje". To znači da se posebni URI treba tretirati kao potencijalno zlonamjerni ulaz neautentificiranog korisnika. Upravo ta činjenica nije uključena u dio „Security Considerations" RFC preporuke, što je dovelo do nepravilnosti u implementaciji.

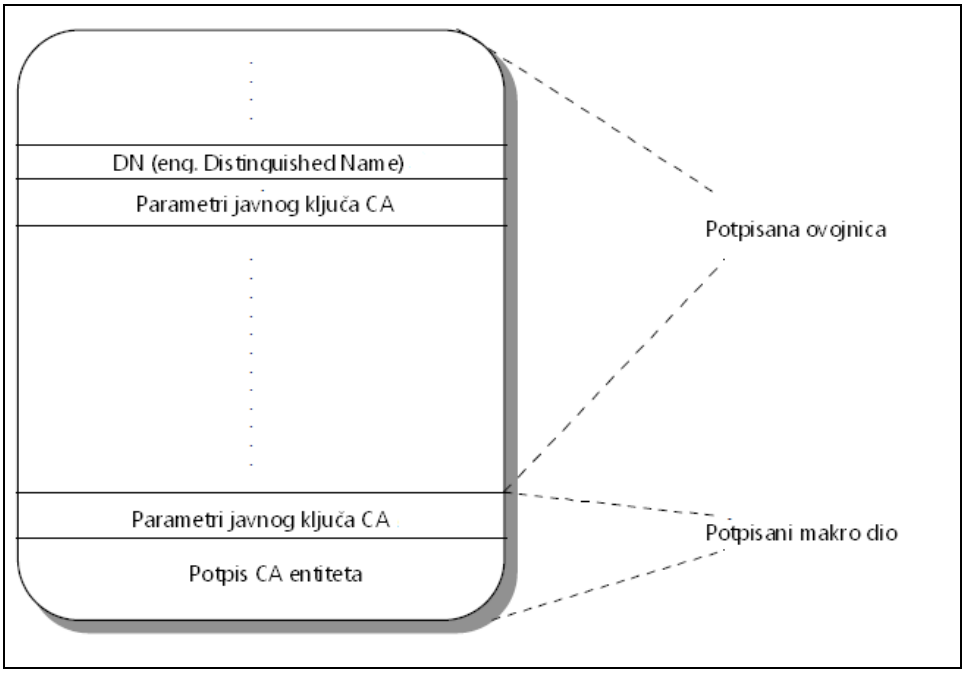
4.3. Supstitucija parametara

Slika 14 prikazuje građu certifikata, koja je pobliže opisana u poglavlju „CA certifikati". Certifikat sadrži elemente koji spadaju u potpisani dio (ime entiteta koji potpisuje certifikat, ime krajnjeg entiteta i javni ključ te dodatne parametre). Potpisani dio može opcionalno sadržati parametre javnog ključa entiteta koji izdaje certifikat, a makro dio obično sadrži i digitalni potpis. Uključivanje parametara javnog ključa u potpisani dio omogućava provjeru potpisa temeljenu na tim parametrima bez potrebe dekriptiranja certifikata. Parametri certifikata entiteta koji izdaje certifikat uključeni su u potpisani dio kako bi CA mogla imati različite parametre javnog ključa. Polje s parametrima javnog ključa krajnjeg entiteta omogućava entitetu da ima različite parametre u odnosu na certifikat koji mu je izdan.



Slika 14. Struktura CA certifikata

Na slici 15 prikazana je struktura CLR popisa, koji sadrži povučene certifikate. Sadržaj CLR popisa sadrži DN (eng. distinguished name) imena certifikata entiteta koji su ih izdali u potpisanom dijelu ovojnice. Opcionalno taj dio može sadržavati parametre javnog ključa CA entiteta. Potpisani makro dio uvijek sadrži digitalni potpis.



Slika 15. Struktura CLR zapisa

4.3.1. Opis problema

Uporaba polja s parametrima javnog ključa entiteta koji izdaje certifikat može se omogućiti napad supstitucijom parametara. Javni ključ i njegovi parametri potrebni su kako bi se provjerio potpis certifikata i CLR zapisa. Javni ključ izdavača se dohvaća preko niza povjerljivih i autentificiranih postupaka te nije dostupan u potpisnim dijelovima certifikata i CLR zapisa. Uporaba digitalnog potpisa pruža određeni stupanj sigurnosti, koji je definiran kao složeni računski potpis ili računanje privatnog ključa (za javni ključ ili parametre određene kvalitete i veličine).

Na primjer, poznato je da su u DSS (eng. Digital Signature Standard) standardu veličina najvećeg modula *p*, veličina malog modula *q* te obilježja *p, p-1* i *q* kritična za sigurnost. Obilježja uključuju osiguravanje da su *p* i *q* primari odgovarajuće veličine, a *q* se dijeli u *p-1*.

Ukratko, postoje dvije lokacije u kojima se pojavljuju parametri javnog ključa, a koje nisu autentificirane. To je slučaj i kada su parametri smješteni u potpisanu ovojnicu, a uzrokovano je činjenicom da su vrijednosti parametra u certifikatu korištene prilikom provjere istog certifikata. Zahvaljujući tome napadač može uvijek zamijeniti navedene parametre.

5. Primjeri iskorištavanja propusta

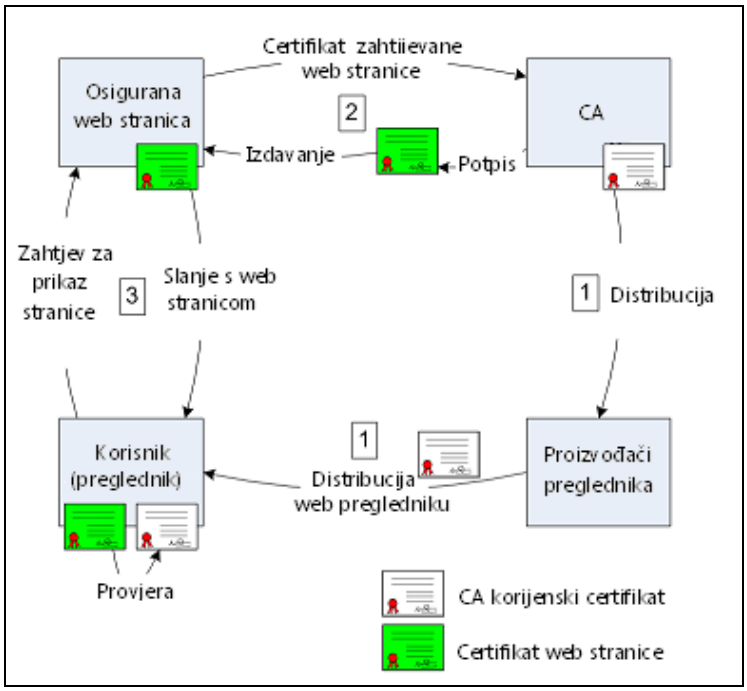
5.1. Teorijski primjeri

5.1.1. Iskorištavanje MD5 problema

Scenarij napada započinje tako da legitimna web stranica zahtjeva certifikat od CA entiteta, kojem vjeruju svi osnovni web preglednici. Budući da je zahtjev legitiman, CA potpisuje certifikat i vraća ga web stranici. Potrebno je izabrati CA koji koristi MD5 *hash* funkcije za generiranje potpisa certifikata, jer je certifikat posebno oblikovan kako bi izazvao koliziju sa drugim certifikatom. Taj drugi certifikat nije certifikat web stranice, nego certifikat središnjeg CA entiteta, koji je moguće iskoristiti za potpisivanje certifikata proizvoljnih web stranica. Budući da su MD5 *hash* vrijednosti legitimnog i lažnog certifikata jednake, digitalni potpis CA entiteta može se jednostavno kopirati u lažni certifikat. Takav certifikat se prikazuje valjanim svim osnovnim web preglednicima.

Slika 16 prikazuje dijagram uporabe certifikata na web stranicama, koji obuhvaća slijedeće korake:

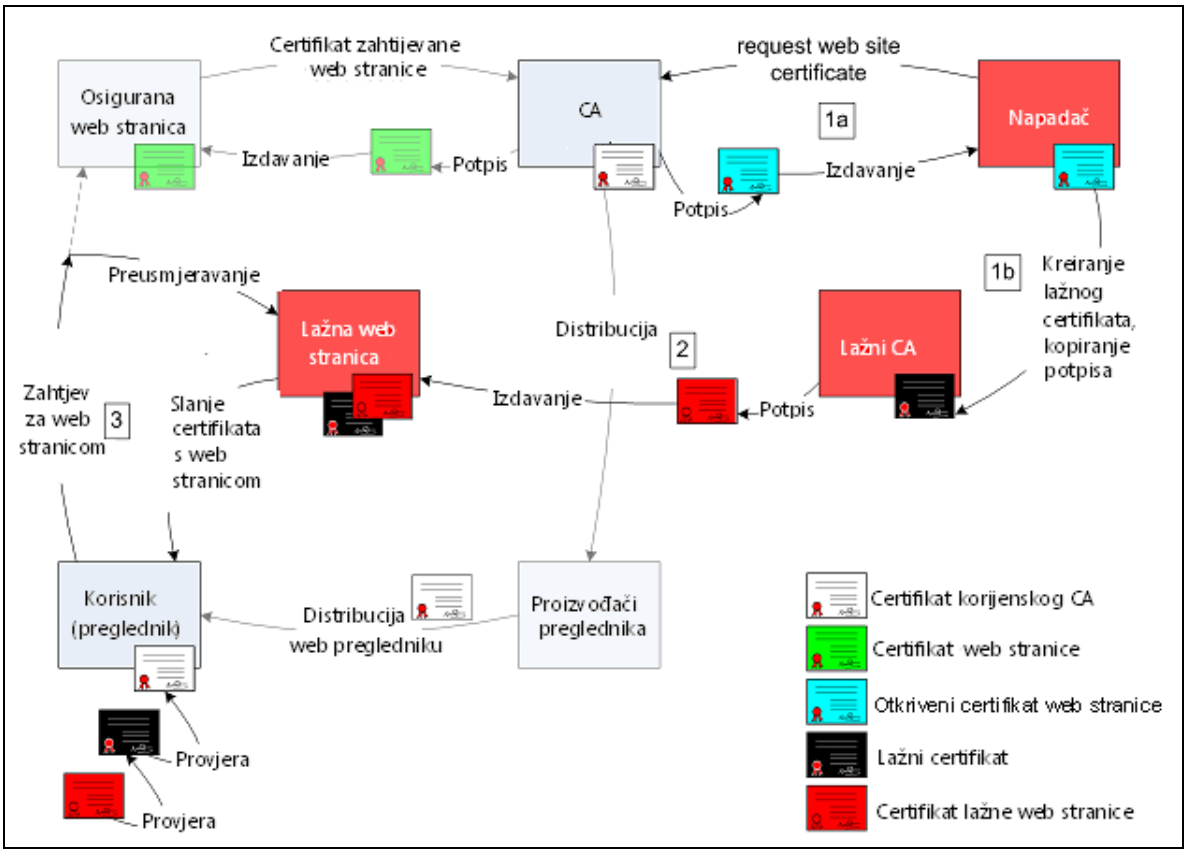
1. CA distribuira certifikate korijenskih CA entiteta (tzv. korijenske CA certifikate) preglednicima i oni se pohranjuju u listu povjerljivih certifikata korisničkog računala. To znači da svi certifikati koje izda takav CA moraju biti prihvaćeni na tom računalu.
2. Organizacije koja želi osigurati svoju web stranicu nabavlja certifikat kod CA entiteta. Taj certifikat je potpisao CA entitet i on garantira za identitet korisnika.
3. Kada korisnik želi posjetiti osiguranu web stranicu, web preglednik prvo zatraži certifikat. Ako se potpis može provjeriti preko certifikata zapisanih na popisu računala, certifikat se prihvaća. Web stranica se učitava u preglednik, a sav promet između preglednika i web stranice osiguran je uporabom kriptografije.



Slika 16. Postupak uporabe certifikata

Slika 17 prikazuje postupak napada, koji uključuje slijedeće korake:

1. otkrivanje legitimnog certifikata web stranice,
2. kreiranje lažnog CA certifikata s istim potpisom kao certifikat legitimne web stranice. Certifikat web stranice nosi izvorni identitet web stranice ali je kreiran drugi javni ključ i potpisan je lažni certifikat. Gradi se kopija izvorne web stranice, postavlja na drugi poslužitelj te opskrbi s lažnim certifikatom.
3. Kada korisnik želi posjetiti osiguranu web stranicu, web preglednik potraži izvornu web stranicu. Obavlja se preusmjeravanje komunikacije na lažnu web stranicu. Ta stranica prikazuje svoj certifikat korisniku, zajedno s lažnim CA certifikatom. Potpis u lažnom CA certifikatu može biti provjeren s lažnim CA certifikatom. Lažni CA certifikat prihvaćaju svi preglednici, budući da njegov potpis (tj. potpis legitimnog certifikata koji se izmijenio) može biti provjeren s certifikatom korijenskog CA entiteta.



Slika 17. Scenarij napada

5.1.2. Iskorištavanje HTTP nedostatka

Ako se sustav implementira prema uputama u preporuci, neautentificirani korisnik može ugraditi proizvoljnu URI adresu unutar certifikata. Na taj način može prisiliti entitet koji vrši provjeru na slanje proizvoljnih HTTP zahtjeva (npr. mreži koja je formalno nedostupna napadaču). Odgovor se ne prosljeđuje napadaču, pa je on ograničen na tzv. *blind* napad. Posebni oblik ovog napada omogućava otkrivanje informacija o entitetu koji vrši provjeru (npr. da li rukuje s porukom elektroničke pošte ili dokumentom). Budući da više URI adresa može biti uključeno u jedan certifikat, teorijski je moguće otkriti informacije o mreži. Za to napadač treba kreirati certifikat s po jednom URI adresom:

- koju sam kontrolira za napad,
- koju sam kontrolira, a služi za mjerenje vremenske udaljenosti između dva pristupa URI adresom koju kontrolira napadač.

Slični problemi se javljaju s specificiranjem nekih drugih parametara koji sadrže URI adrese:

- CPS pokazivači (eng. CPS Pointer),
- CRL dijeljene točke (eng. CRL Distribution Points),
- Informacije o pristupu OCSP (eng. Authority Information Access OCSP).

CRL dijeljene točke i OCSP URI adresama pristupa se samo jednom nakon što je provjera certifikata uspješna. Iako to smanjuje broj korisnika koji mogu zlouporabiti problem, ipak donosi određene sigurnosne rizike.

Ranjive aplikacije:

- Microsoft Outlook,
- Windows Live Mail,

- Microsoft Office 2007.

5.1.3. Iskorištavanje supstitucije parametara

Ako se koriste parametri javnog ključa u potpisanim dijelovima, napadač koji želi zamijeniti, izmijeniti ili kreirati lažne certifikate i CRL zapise može izvršiti supstituciju parametara. Napadač mora zamijeniti vrijednosti u objektima (certifikat i CRL zapis) te ih ponovno potpisati. To mu omogućava prevođenje teškog kriptografskog problema u problem pronalaženja novog skupa parametara i javnog ključa koji odgovara valjanom javnom ključu. Postupak pronalaženja takvog odgovarajućeg ključa ima varijabilnu složenost, a ovisi o matematičkoj složenosti kriptografskog postupka.

Zlouporabom opisanog nedostatka može se ugroziti sigurnost cijele PKI infrastrukture budući da napadač može modificirati ili kreirati lažne certifikate i CRL zapise za središnje CA entitete u nizu i za krajnje entitete. To je moguće jer se povjerenje u PKI infrastrukturi zasniva na autentifikaciji certifikata i CRL zapisa.

5.2. Praktični primjeri

Većina opisanih problema samo je prikazana u teoriji, dok praktični primjeri iskorištavanja nisu još ostvareni. Što se tiče napada na certifikate tijekom povijesti, moguće je pratiti problem s MD5 algoritmom. Godine 2005. Arjen Lenstra, Benne de Weger i Xiaoyun Wang izveli su napad, koji pokazuje mogućnost uzrokovanja kolizije certifikata. Dvije godine kasnije, izveden je složeniji napad koji iskorištava probleme MD5 algoritma. Napad su ostvario Marc Stevens, a nazvao ga je kolizija biranim prefiksom. Pred kraj prošle godine, skupina stručnjaka objavila je kako je uspjela kreirati lažni certifikat koji većina preglednika prihvaća kao valjan.

6. Metode zaštite

6.1. Vatrozid

Vatrozid (eng. firewall) je uređaj koji pomaže u provođenju obrane od zlonamjernog prometa u mrežama, a predstavlja poveznicu između sigurne i nesigurne mreže.

Može se koristiti za zaštitu:

- Internet mreže,
- intranet mreže,
- posebnih mreža,
- poslužitelja i
- korisničkih računala.

Osnovna funkcija vatrozida je filtriranje svog prometa koji ulazi u mrežu ili izlazi iz mreže, u svrhu provjere valjanosti. Temelji rad na provjeri paketa pomoću kriterija koje postavlja korisnik ili administrator.

Vatrozid ne pruža potpunu zaštitu sustava od napadača (npr. napadači mogu pronaći način za podizanje ovlasti na sustavu), jer je nemoguće zabraniti svim komponentama pristup Internetu. Zbog toga definirane je DMS (eng. demilitarized zone) zona gdje se smještaju uređaji koji moraju biti vidljivi na Internetu. Prometom se tada može upravljati dodatnim sigurnosnim proizvodima poput:

- IDS (eng. intrusion detection systems) sustava temeljenih na mrežama,
- IDS sustava temeljenim na poslužiteljima.

Uporaba vatrozida (i drugih sličnih programa) igra važnu u odbacivanju onih stranica koji nemaju ispravne potpise i certifikate čime se značajno smanjuje napadačev manevarski prostor.

6.2. VPN

VPN (eng. Virtual Private Networking) omogućava organizacijama uporabu javnih mreža za spajanje na zaštićene resurse lokalne mreže, kao da su izravno priključeni u lokalnu mrežu. Sve VPN mreže

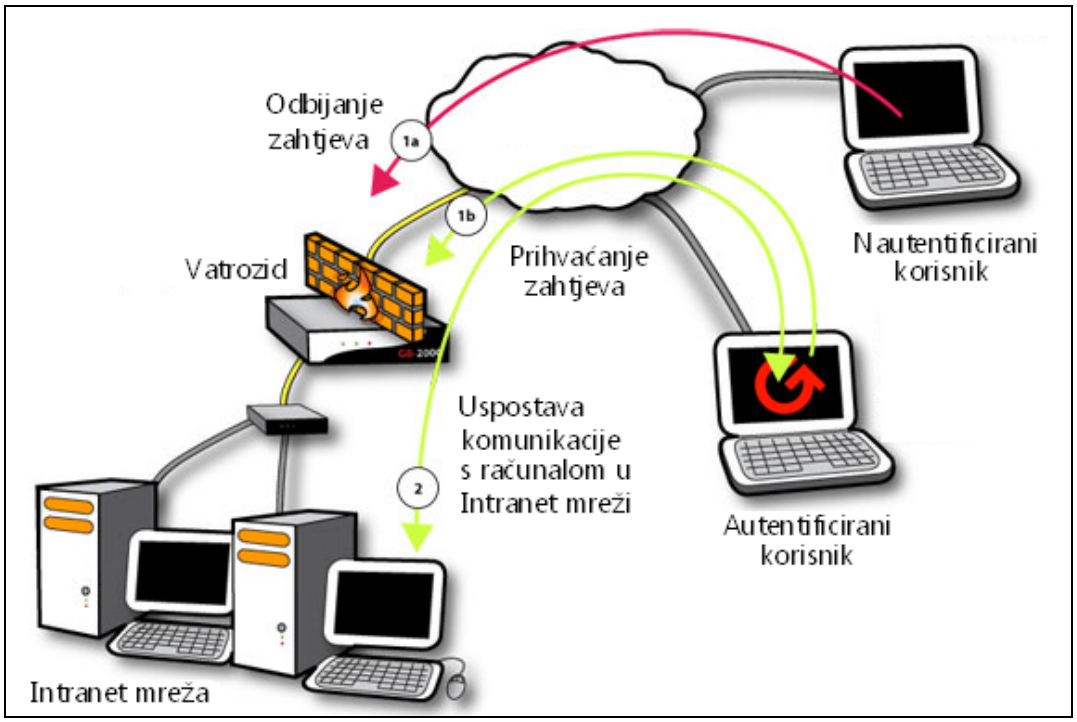
funkcioniraju na slijedeći način: promet je kriptiran i umetnut u nove pakete te prenijet uređajima u intranet mreži. Nakon primanja takvih paketa, uređaji ih enkapsuliraju, provjeravaju integritet i dekriptiraju pakete te ih šalju odredištu.

Uporaba VPN mreža omogućava organizacijama korištenje jeftinijih, dijeljenih uređaja poput modema, DSL-a i sl. Također, VPN tunel skriva sav promet koji putuje kroz njega, kriptira ga, skriva identitet i protokol kojim uređaji komuniciraju.

Omogućava implementaciju bilo kojeg protokola, te pomaže pri zaštiti informacija i smanjenju opasnosti njihovog otkrivanja. Ipak VPN tehnologija sadrži određene rizike poput mogućnosti krađe sjednica i povećanja prava pristupa mrežnim resursima. Kako bi se smanjila opasnost od navedenog rizika moguće je implementirati vatrozid i IDS programe (slika 18).

Također, svaki tunel do intranet mreže može predstavljati sigurnosni rizik ako se s njim ne upravlja ispravno. Unatoč tome, VPN mreže postaju sve popularnije i raširenije.

Korištenje VPN mreža omogućava bolju zaštitu intranet sustava (tj. servisa u lokalnoj mreži). Podaci se mogu sigurno razmjenjivati Implementacijom VPN mreže, korisnicima je moguće osigurati udaljeni pristup mreži dodjelom digitalnih certifikata. Svaki korisnik ima odgovarajuća prava pristupa podacima te njihovoj izmjeni, dodavanju I brisanju. U slučaju da nekom korisniku treba zabraniti pristup iz nekog razloga (zlouporaba prava, napuštanje radnog mjesta) njegov certifikat se dodaje na CRL popis. Budući da je certifikat korisnika povučen on više nema pristup intranet sustavu, dok ostali korisnici mogu nesmetano koristiti resurse.



Slika 18. VPN i vatrozid

6.3. Antivirusni programi

Uporaba antivirusnih programa je najbolji način zaštite od virusa, crvi i trojanskih konja. Danas postoje razni besplatni i komercijalni alati za zaštitu sustava, a temelje se na analizi prometa u svrhu otkrivanja zlonamjernih programa. Većina alata osim prepoznavanja virusa i ostalih štetnih programa, ima i funkcionalnost njihova uklanjanja.

Nakon otkrivanja nekog zlonamjernog programa, moguće je primijeniti antivirusni alat specijaliziran za njegovo uklanjanje. Antivirusni programi ne mogu otkriti izvor virusa, ali mogu detektirati njegovu prisutnost na sustavu.

Za postizanje što bolje zaštite sustava i mreža preporuča se kombiniranje raznih antivirusnih alata.

Kod PKI infrastrukture važnost antivirusnih alata se ističe u slučaju kada napadač želi podmetnuti korisniku određeni zlonamjerni kod. Tada se sustav može obraniti i nakon prihvaćanja određenog virusa. Napadač može iskoristit nedostatke PKI infrastrukture (npr. MD5 problem) te kreirati ispravan certifikat za lažnu web stranicu. Na toj web stranici može smjestiti posebno oblikovan programski kod, koji ce se pokrenuti kada korisnikov preglednik prihvati zahtjev za pregledom stranice. Preglednik prihvaća zahtjev jer vjeruje lažnom certifikatu. Kada se zlonamjerni program pokrene, jedini način zaštite je primjena odgovarajućeg antivirusnog programa.

6.4. Ostale metode zaštite

U ovom poglavlju opisane su metode zaštite za pojedini opisani nedostatak kod PKI infrastrukture:

1. Kako bi se riješio problem MD5 nedostatka, stručnjaci savjetuju prestanak uporabe MD5 algoritma kako bi se spriječilo iskorištavanje njegove ranjivosti. Kao zamjenu za navedeni algoritam preporuča se uporaba SHA-1, SHA-2 i sličnih algoritama. Većina CA entiteta već poduzima takve mjere, a najbolja je praksa potpuno izbaciti MD5 algoritam iz uporabe. U slučaju kada se ne može izbjeći uporaba MD5 algoritma, stručnjaci preporučuju uvođenje slučajnog generiranja serijskih brojeva svih novo izdanih certifikata. Također, stručnjaci ukazuju na potrebu pažljivog uvođenja algoritma SHA-1 te izbjegavanje njegove primjene u novim proizvodima jer je teorijski prikazano da je ranjiv na iste probleme kao i MD5 algoritam. Zbog toga, za sada se preporuča prelazak na SHA-2 algoritam.
2. Što se tiče HTTP problema, nisu dostupni odgovarajući programski alati za zaštitu. Blokiranje određenog mrežnog prometa (npr. uporaba vatrozida) je jedina moguća opcija zaštite. Time se odbacuje promet koji ne dolazi s povjerljivih web odredišta. Također, moguće je provoditi filtriranje izlaznog prometa na posredničkim ili aplikacijskim pristupima. Filtriranjem izlaznog prometa može se spriječiti proizvoljno slanje zahtjeva.
3. Zaštita od supstitucije parametara može se provesti na slijedeće načine:
    a. Analiza parametara – potrebna jer X.509 standard provodi fleksibilne mehanizme za registraciju ključa i njegovih parametara za razne vrste algoritama. Prilikom registriranja potrebno je u potpunosti analizirati problem supstitucije parametara. Ako se pokaže da je supstituciju teško provesti, parametri mogu biti registrirani kao dio parametara javnog ključa. U suprotnome, oni trebaju biti registrirani kao dio javnog ključa.
    b. Ignoriranje polja za parametre javnog ključa izdavača certifikata – za sustave poput DSS-a parametri su već registrirani, a analiza pokazuje da je supstitucija jednostavnija od računanja diskretnog logaritma. Tada parametri javnog ključa trebaju biti zanemareni.
    c. Promjena registriranja – provodi se također za sustave poput DSS, a zahtjeva izmjenu registriranja kako polje parametara ne bi prenosilo parametre. Oni se mogu prenositi opcionalno u informacijskom polju javnog ključa krajnjeg entiteta.
    d. Korištenje parametara u polju parametara javnog ključa krajnjeg entiteta – prijašnja rješenja ne smanjuju mogućnost da različiti korisnici posjeduju različite parametre. U niz certifikata i CRL zapisa proizvoljne duljine moguće je izbjeći napad supstitucijom ako se zadovolje slijedeći uvjeti. Certifikati i CRL zapisi trebaju započeti s autentificiranim javnim ključem i parametrima te koristiti vrijednosti u polju parametara javnog ključa krajnjeg entiteta.
    e. Provjera kvalitete i veličine parametara – tijekom uporabe certifikata ili CRL zapisa treba provjeriti kvalitetu i veličinu neautentificiranih parametara.

7. Zaključak

Kriptografija ima vrlo važnu ulogu u zaštiti informacija koje se prenose preko Internet mreže. To je pogotovo izraženo u današnje vrijeme, kada se raznim osjetljivim informacijama rukuje preko mreže ili su one pohranjene na poslužiteljima i u bazama podataka. Kako bi se osigurala tajnost i neizmijenjenost podataka te autentičnost pošiljatelja i primatelja, potrebno je razvijati složene algoritme kriptiranja. Implementacija PKI infrastrukture zadovoljava osnovne zahtjeve o sigurnosti osjetljivih podataka. Uporaba javnih i privatnih ključeva, kao i digitalnih potpisa, omogućava lako upravljanje prihvaćenjem legitimnog i blokiranjem neprihvatljivog prometa.

Budući da se računarski svijet razvija velikom brzinom, potrebno je održavati odgovarajuću razinu složenosti kriptografskih sustava kako bi se spriječilo njihovo ugrožavanje. Primjer za to je upravo nedostatak kod PKI infrastrukture, koja se temelji na MD5 algoritmu. Iako je u prošlosti smatrano da navedeni algoritam zadovoljava kriptografske potrebe, razvojem boljih tehnologija, prikazano je kako je moguće ugroziti njegovu sigurnost.

Kako bi se osigurala sigurnost razmjene informacije (poruka elektroničke pošte, osobnih podataka, financijskih podataka i sl.) nije dovoljno samo implementirati PKI infrastrukturu prema odgovarajućim standardima. Potrebno je poduzeti mjere zaštite sustava, poput primjene vatrozida i antivirusnih programa. Osim toga, nakon implementacije PKI infrastrukture moguće je poduzeti dodatne mjera kako bi se izbjegli osnovni napadi.

Daljnjim napretkom računala i tehnologija očekuje se pojava novih tehnika probijanja postojećih mehanizma zaštite. Iako stručnjaci već ukazuju na ranjivosti MD5 algoritma i potiču na prestanak njegove uporabe, vjerojatno će se pronaći još neki nedostaci i metode njihove zlouporabe (npr. desktop probijanje MD5 CA certifikata). Zbog toga, svakodnevno se radi na razvoju i poboljšanju sigurnosnih algoritma pa se u budućnosti može očekivati uporaba novih sustava kriptiranja.

8. Reference

[1] Kriptiranje, http://en.wikipedia.org/wiki/Encryption, veljača, 2009.

[2] Kriptiranje, http://computer.howstuffworks.com/encryption.htm, veljača, 2009.

[3] Joel Weise, Public Key Infrastructure Overview, http://www.sun.com/blueprints/0801/publickey.pdf, kolovoz, 2001.

[4] PKI, http://en.wikipedia.org/wiki/Public_key_infrastructure, veljača, 2009.

[5] X.509, http://en.wikipedia.org/wiki/X.509, veljača, 2009.

[6] X.509, http://msdn.microsoft.com/en-us/library/aa480610.aspx, veljača, 2009.

[7] Arhitektura PKI, http://tools.ietf.org/html/draft-ietf-pkix-apki-00#page-16, veljača, 2009.

[8] PKI komponene, http://www.opengroup.org/onlinepubs/009219899/chap2.htm, veljača, 2009.

[9] Internet X.509 Public Key Infrastructure, http://www.ietf.org/rfc/rfc2459.txt, veljača, 2009.

[10] Digitalni potpisi, http://en.wikipedia.org/wiki/Digital_signature, veljača, 2009.

[11] Alexander Sotirov, Marc Stevens, Jacob Appelbaum, Arjen Lenstra, David Molnar, Dag Arne Osvik, Benne de Weger: Creating a rogue CA certificate, http://www.win.tue.nl/hashclash/rogue-ca/, prosinac, 2008.

[12] Marc Stevens, Arjen Lenstra, and Benne de Weger: Chosen-prefix Collisions for MD5 and Colliding X.509 Certificates for Different Identities, http://www.win.tue.nl/hashclash/EC07v2.0.pdf, 2007.

[13] Colliding X.509 Certificates for Diferent Indetities, http://www.win.tue.nl/hashclash/TargetCollidingCertificates/, veljača, 2009.

[14] HTTP over X.509, https://www.cynops.de/techzone/http_over_x509.html, veljača, 2009.

[15] Santosh Chokhani: A Security Flaw in the X.509 Standard, http://csrc.nist.gov/nissc/1996/papers/NISSC96/paper075/paper.pdf, veljača, 2009.

[16] Vatrozid, http://en.wikipedia.org/wiki/Firewall_(networking), veljača, 2009.

[17] VPN, http://www.oreillynet.com/pub/a/security/2004/09/23/vpns_and_pki.html, veljača, 2009.

[18] Antivirusni programi, http://en.wikipedia.org/wiki/Antivirus_software, veljača, 2009.