Digitalni podpisi

Jan Božič

20.12.2020

Uvod

Preden sploh začnemo govoriti o digitalnih podpisih bi rad naredil razločitev med nekaj zrari ki precej sovpadajo z njimi, da kasneje ne bo nesporazumov.

- **Asimetrična kriptografija:** veja kriprografije ki se ukvarja s funkcijami/algoritmi ki producirajo dva različna ključa za enkripcijo podatkov; ’privatnega’ in ’javnega’, en za dekripcijo, drugi za enkripcijo.
- **Elektronski podpis:** kakšen koli podatek v elektronski obliki, ki je logično asociiran z nekim drugim podatkom elektronske oblike. V primeru digitalnih podpisov je to podatek generiran ko s privatnim ključem ’podpišemo’ neke druge podatke.
- **Digitalni podpis:** je matematična shema oz. implementacija le tega ki omogoča preverjanje avtentičnosti podatkov. Dandanes uporablja asimetrično kriptografijo da podpiše podatek ki ga je kaneje možno preveriti.
- **Digitalni certifikat:** je elektronski dokument za dokaz lastništva javnega ključa.

Teoretična definicija

Shema digitalnega podpisa je v osnovi sestavljena iz sledečih dreh algoritmov:

- Generator ključev (key generator), pri kterem je pomembno da generira ključe naključno z enakomerno razporeditvijo. Generira tako privatni kot javni ključ.
- Algoritem za podpisovanje (signing algorithm), ki s podanim privatnim ključem in sporočilom generira podpis.
- Algoritem za preverjanje (*signature verifying* algorithm), ki s sporočilom, podpisom in javnim ključem preveri če se podpis ujema s sporočilom.

Zadovoljene morajo biti tri lastnosti. Prvo, avtentičnost podpisa, ki je generiran iz (nespreminjajočega) sporočila in privatnega ključa mora biti vedno preverljiva s pripadajočim javnim klučem. Kot drugo, generiranje podpisa mora biti računsko nemogoče brez posedovanja privtnega ključa. In kot tretje mora biti pravtako računsko nemogoče generirati privatni ključ.

Pravilna avtentikacija preko digitalnega podpisa je sledeča:

1. Pošiljatelj naredi sporočilo ki ga želi poslati.
2. Pošiljajoča programska oprema generira podpis s privatnim ključem pošiljatelja.
3. Podpis je dodan na konec sporočila.
4. Prejemnikova programska oprema uporabi sporočilo in javni ključ da ponovno generira sporočilo, ter ga nato primerja z originalnim sporočilom da zagotovi avtentičnost.

Kar je pomembno poudariti je da v takem primeru pridemo do kar nekaj omejitev. Sporočilo je omejeno na dolžino, pravtako pa je asimetrična kriptografija v splošnem računsko zahtevnejša od simerišne. Zato se v praksi vedno uporablja digitalne podpise skupaj z hash algoritmi, ki najprej generirajo hash sporočila, kar je potlej tisti podatek ki ga potlej podpišemo. Ko pa preverjamo pristnost sporočila najprej generiramo en hash iz sporočila in enega iz podpisa ter ju primerjamo.

Nameni uporabe

Digitalni podpisi imajo danes zelo širok namen uporabe, od kriptiranja emailov do omogočanja posameznikom in podjetjem da se varno vpišejo in uporabljajo spletne storitve. Ampak skoraj vse lahko razdelimo v štiri podskupine, avtentikacija, integriteta, nezatajljivost in anonimna komunikacija:

Avtentikacija

Digitalne podpise lako uporabimo za preverjanje prave identitete pošiljatelja sporočil. Pri tem se pogosto uporabljao tudi digitalni certifikati, ki so lahko prav tako digitalno podpisani v nekaterih primerih s strani certifikatne avtoritete (model znan kot Public key infrastructure ali PKI), v drugi pa s strani kogarkoli drugega ki je dovolj prepričan da jamči da certifikat res pripada tistemu za katerega certifikat trdi da mu pripada (midel znan kot Web of Trust ali WOT).

Integriteta

Velikokrat so sporočila tako pomembna da želimo biti prepričani da sporočilo ni bilo (zlonamerno) spremenjeno med prenosom ali medtem ko je bilo hranjeno. Čeprav enkripcija postkrbi da ne moremo prebrati sporočila, v nekaterih primerih še vedno lahko spremenimo njegovo vsebimo. Če pa je sporočilo podpisano z digitalnim podpisom, bo kakršnakoli sprememba zaznana ko bo podpis preverjen.

Nezatajljivost

Oziroma bolj natančno nezatajljivost izvora. Ta lastnost je neposrednja posledica prejšnjih dveh skupaj in mogoče njabolj pomembna. Tisti ki je podpisal sporočilo kasneje ne more trditi da sporočila ni podpisal. Pravtako pa nihče drug ne more ponarediti podpisa, tudi če poseduje javni ključ. Ker so dandanes digitalni podpisi skoraj povsod po svetu pravno zavezujoči, si zlahka lahko prestavljamo morebiten vpliv na sodne odločitve.

Anonimna komunikacija

V teoriji bi jih lahko uporabljali za anonimno komunikacijo skozi celotno sejo pogovora, vendar se v praksi po večini uporabljajo le na za začetno vzpostavitev anonimne komunikacije, saj postane postopek precej enostavnejši. Le portebno je genetirati ključ s simetrično enkripcijo ki ga želimo uporabiti med sejo, nato pa ga z javnim ključem prejemnika zakriptiramo in mu ga pošljemo. Če je prejemnik res edini ki poseduje privatni ključ, potlej je tudi edini ki lahko dešifrira ključ za sejo ga lahko že začne uporabljati.

Varnostne konsideracje

Hramba privatnih ključev

Vse sistemi ki uporabljajo koncept javnih in privatni ključev se zanašajo da bo privatni ključ res ostal skrivnost. Ključ je lahko shranjen na računalniku kjer je dostopen le z geslom, ampak to ima dva problema:

- Uporabnik lahko podpisuje sporočila le na tistem računalniku.
- Varnost ključa je odvisna od varnosti računalnika.

Bolj varna alternativa je hramba privatnega ključa na pametni kartici. Pametne kartice so svoj mali računalnik, in veliko jih je narejenih tako da so kar se da zaščitene pred posegi izven njihovega primarnega namena. Ko shranimo privatni ključ na kartico s katere ga ne moremo preprosto sneti, uporabljamo kartico da generira digitalni podpis željenega sporočila. Seveda je pravtako možna uporaba gesla ali PIN-a, za primer kraje ali izgube kartice, pravtako pa lahko uporabimo kartico na različnih napravah.

Uporaba le na aplikacijah ki jim zaupamo

Glavna stvar ki loči digitalni podpis od fizičnega je to da oseba ki podpisuje v večini primerov ne vidi neposredno kaj podpisuje. Vse kar dejansko vidi je interpretacija sporočila ki ga bo podpisal in zato mora zaupati aplikaciji da bo resnično podpisala predstavljeno interpretacijo. Če je bila recimo v aplikacijo podtaknjena zlonamerna koda lahko zelo enostavno podpiše sporočila napadalca medtem ko uporabniku zatrdi da je podpisano le tisto kar uporabnik pričakuje.

Ta problem je možno rešiti tako da med aplikacijo in digitalni podpis vrinemo dodaten korak avtentikacije s predogledom sporočila ki bo podpisano.

WYSIWYS

WYSIWYS kratica stoji za What You See Is What You Sign. Pomeni da semantična interpretacija sporočila ne sme biti spremenjena, malo manj očitno pa tudi pomeni da sporočilo ne sme vsebovati informacij skritih pred tistim ki sporočilo podpisuje. WYSIWYS je velikokrat pogoj za veljavnost digitalnega podpisa, ampak ta zahteva ni vedno enostavna za izpolniti. Ker tehnično gledano ko podpišemo sporočilo podpišemo surove enke in ničle in ne samega semantičnega pomena, ki je interpretiran posebej z drugo aplikacijo. V primeru da se spremeni način interpretacije v aplikaciji, zaradi posodobitev recimo, potlej se spremeni tudi pomen sporočila. To lahko posledično pripeje do zmede kaj točno je bilo mišljeno ko je bilo sporoćilo podpisano. Pravtako je pametno premisliti kaj točno podpisujemo. Neposredno sporočilo samo ali pa le recimo spletno povezavo do sporočila. V slednjem primeru se moramo zavedati da se lahko sporočilo na spletni povezavi spremeni breez da vi to razveljavilo digitalni podpis.

Standardi danes

Zaradi široke uporabe obstaja veliko standardev in specifikacij digitalnih podpisov in predvsem certifikatov. SIM kartice v mobilnih telefonih uporabljajo ETSI-MSS specifikacijo, medem ko bančne kartice uporabljajo PCI DSS ki ga je razvila finančna industrija. FIPS 140-2 in FIPS 201 uporablja Ameriška vlada… lako bi šli dalje vendar se bomo tukaj osredotočili na dva največja.

X.509

Ta standard ni nič novega. Predlagan je bil v RFC 2527 leta 98 in sprejet januarja 99, ki je do danes nadomeščen z RFC 5280.

Značilnost X.509 sistema Public key infrastricture ali PKI pri kateri tisti ki si želi ustvariti digitalni certifikat najprej generira svoj par ključev in certifikat z željenimi podatki v katerega doda svoj javni ključ. Nato certifikat pošlje certifikani avtoriteti (certificate authority ali CA, sedaj veste kaj pomeni kratica SIGEN-CA) na overitev in podpis. Temu se reče certificate signing request ali CSR. Privatni kluč vedno obdrži pri sebi.

Standard je precej nespecifičen glede attributov potrebnih v certifikatu, zato omogoča veliko maneverskega prostora za CA da sama določi zahtevane arrtibute. To pa je tudi dvorezen meč ker posledično omeji uporabnost certifikata v več domenah. Kot rešitev je bilo razvitih več PKCS standardov (Public Key Cryptography Standards).

Certifikat od CA je lahko podpisan s strani druge nadrejene CA ali če je na vrhu hirearhije, samopodpisan. Le ta se imenuje root certificate. CA omogoči prost dostop do svojega certifikata, da lahko kdorkoli sam preverja pristnost izdanih certifikatov.

V RFC 5280 so pravtako vključene specifikacije za preklic cerifikatov. Uporablja certificate revocation list ali CRL (na primer: http://si-trust-data.gov.si/crl/sigen-ca.pem). Kasneje pa je bil z enakim namenom specificiran Online Certificate Status Protocol ali OCSP.

Bolj razširjene implementacije: OpenSSL knjižnjica implementira vse potrebno in je na voljo na Windows, Linux in Mac OS-ih.

OpenPGP

Za OpenPGP je zaslužno podjetje PGP Inc. ki je pravtako ustvarilo algoritem RSA. Trenutna specifikacija je RFC 4880.

Za razliko od X.509 ki sloni na PKI, OpenPGP implementira Web of trust ali WOT. Ideja je da so certifikati podpisani večkrat in posledično je dovol da zaupaš le enemu od podpisanih. Posledično čez čas pravtako nabereš več javnih ključev ki im zaupaš in zato še lažje preveriš naslednje certifikate. To poskrbi za decentralizirano mrežo zaupanja po celotnem internetu. To ima svoje prednosti, vendar predpostavlja da bodo uporabniki občasno sami preverjali in podpisovali nove certifikate, vendar ne ponudi mehanizma za inicativo. Zato so v kasnejših verzijah razvili trust signatures, ki pridejo v različnih nivojih in jih je možno uporabljati kot CA indicirajo ne le pristnost ampak tudi da ima lastnik certifikata pravico podpisovati ključe ki so en nivo nižje od njega.

OpenPGP je od začetka omogočal tudi preklic certifikatov, ki je precej podoben CRL-ju. Kasnejše verzije pa podpirajo možnost roka uporabe.

Bolj razširjene implementacije: GnuPG (Gnu Privacy Guard) knjižnjica implementira celoten standard za Linux. Na njej bazirata tudi GPG Tools za Mac in Gpg4win za Windows.

Algoritmi za digitalni podpis

Če naštejemo le nekaj najpopularnejših:

- RSA
- DSA
- ECDSA
- EdDSA
- RSA z SHA
- ELG