Elektronički (napredni) potpis (digitalni potpis)

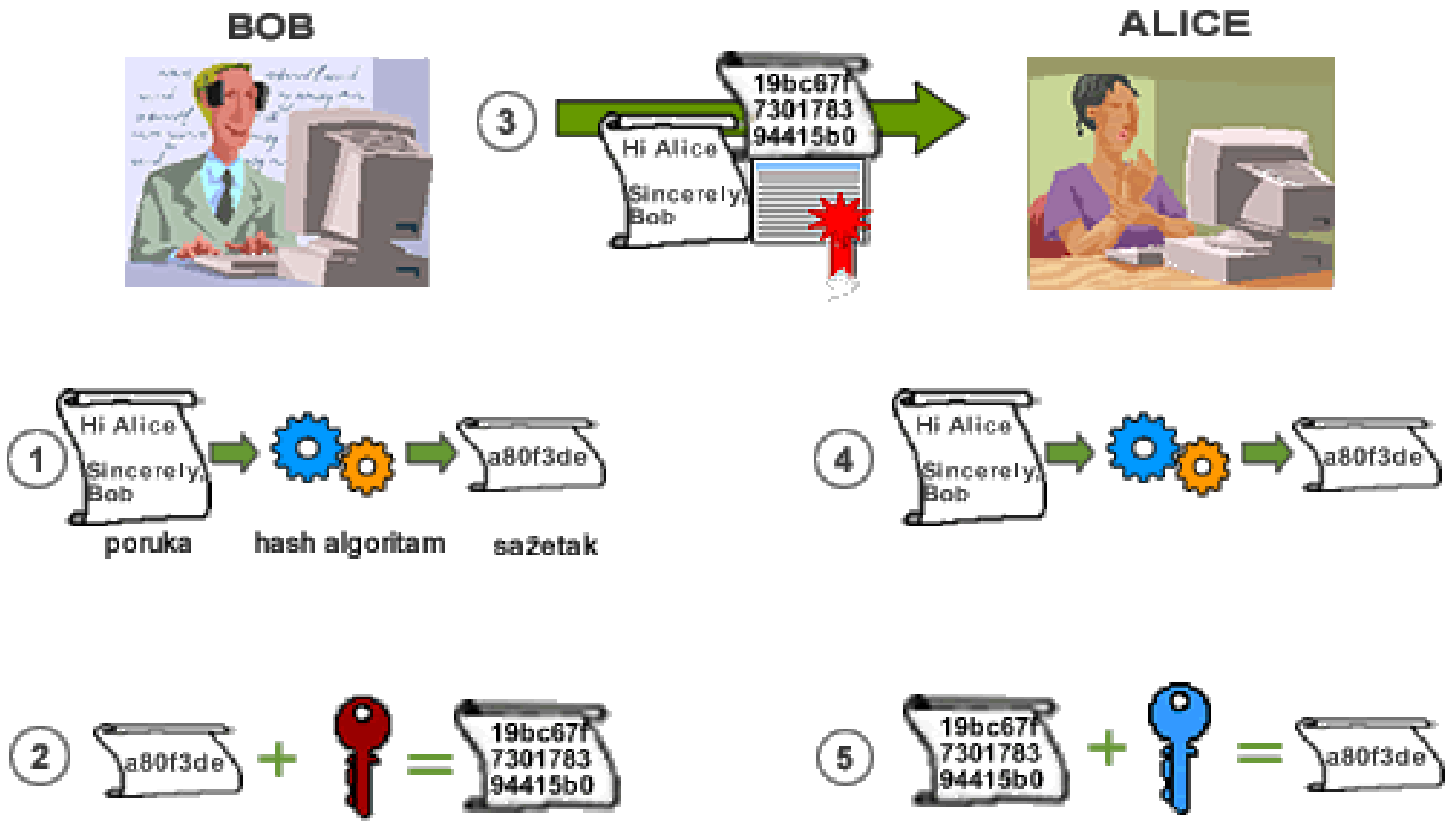
Definicija

- Digitalni potpis - digitalni kod koji služi za zaštitu poruka koje se elektronički prenose putem javne mreže.
- Svrha digitalnog potpisa:
  - omogućiti identifikaciju pošiljaoca
  - osigurati autentičnost sadržaja poruke.

- Upotreba e-potpisa je u RH regulirana Zakonom u elektroničkom potpisu (ZEP) (NN 10/02).
- ZEP definira elektronički potpis kao “skup podataka u elektroničkom obliku koji su pridruženi ili su logički povezani s drugim podacima u elektroničkom obliku i koji služe za identifikaciju potpisnika i vjerodostojnosti potpisanoga elektroničkog dokumenta” [NN, 2010]. Uz elektronički potpis, ZEP definira i napredni elektronički potpis kao “elektronički potpis koji [NN, 2010]:
  1. je povezan isključivo s potpisnikom,
  2. nedvojbeno identificira potpisnika,
  3. nastaje korištenjem sredstava kojima potpisnik može samostalno upravljati i koja su isključivo pod nadzorom potpisnika,
  4. sadržava izravnu povezanost s podacima na koje se odnosi i to na način koji nedvojbeno omogućava uvid u bilo koju izmjenu izvornih podataka.”

- Ako je izrađen u skladu s odredbama Zakona, napredni elektronički potpis ima istu pravnu snagu i zamjenjuje vlastoručni potpis i otisak pečata. Potpuno zamjenjuje potpis na papiru te se njegovim korištenjem uz autentičnost i integritet osigurava i neporecivost dokumenta.

Princip rada digitalnog potpisa

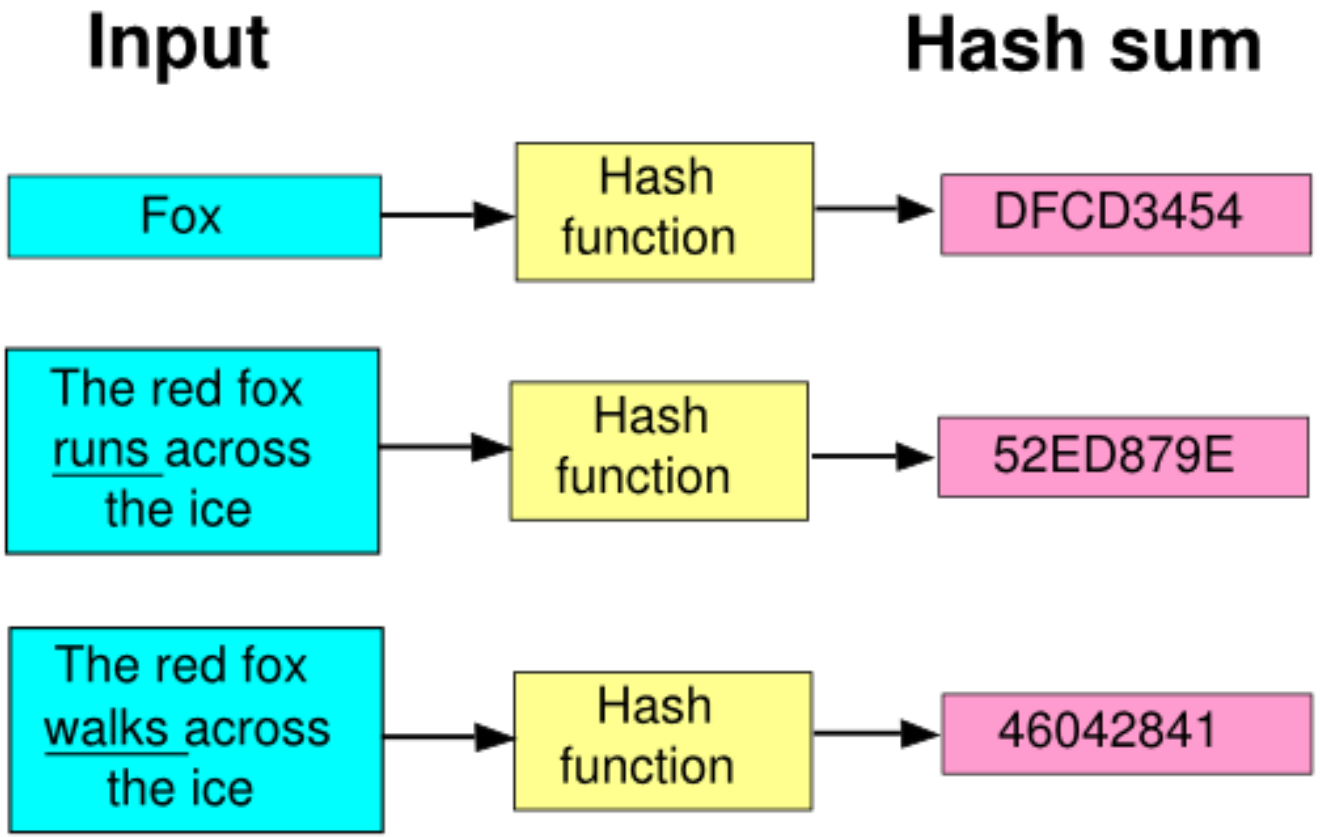


Algoritmi sažimanja - *Hash* funkcija

- Matematičko gledište - funkcija koja transformira proizvoljan broj elemenata ulazne domene u jedan element kodomene
- ICT gledište - algoritam kojim se varijabilni ulaz proizvoljne duljine transformira u niz znakova fiksno određene duljine
- Karakteristike:
  - niz ulaznih podataka je proizvoljne veličine
  - izlazni podatak je stalne veličine
  - nemoguće je izvesti inverznu funkciju
  - ne daje dva ista izlaza za dva različita ulaza

Hash funkcija

- ✓ od ulaza varijabilne veličine vraća znakovni niz fiksne dužine



Hash funkcija

- ✓ Kriptografske hash funkcije su hash funkcije s dodatnim sigurnosnim svojstvima kako bi ih se moglo koristiti za autentifikaciju i očuvanje integriteta podataka
- ✓ Poruka se dijeli na blokove veličine 512 bita
- ✓ MD5 algoritam radi na 128-bitnom izrazu koji se dijeli na 4 32-bitne riječi
- ✓ Zatim se procesiraju redom svi 512-bitni blokovi kojima se mijenja 128-bitni izraz

Hash funkcija

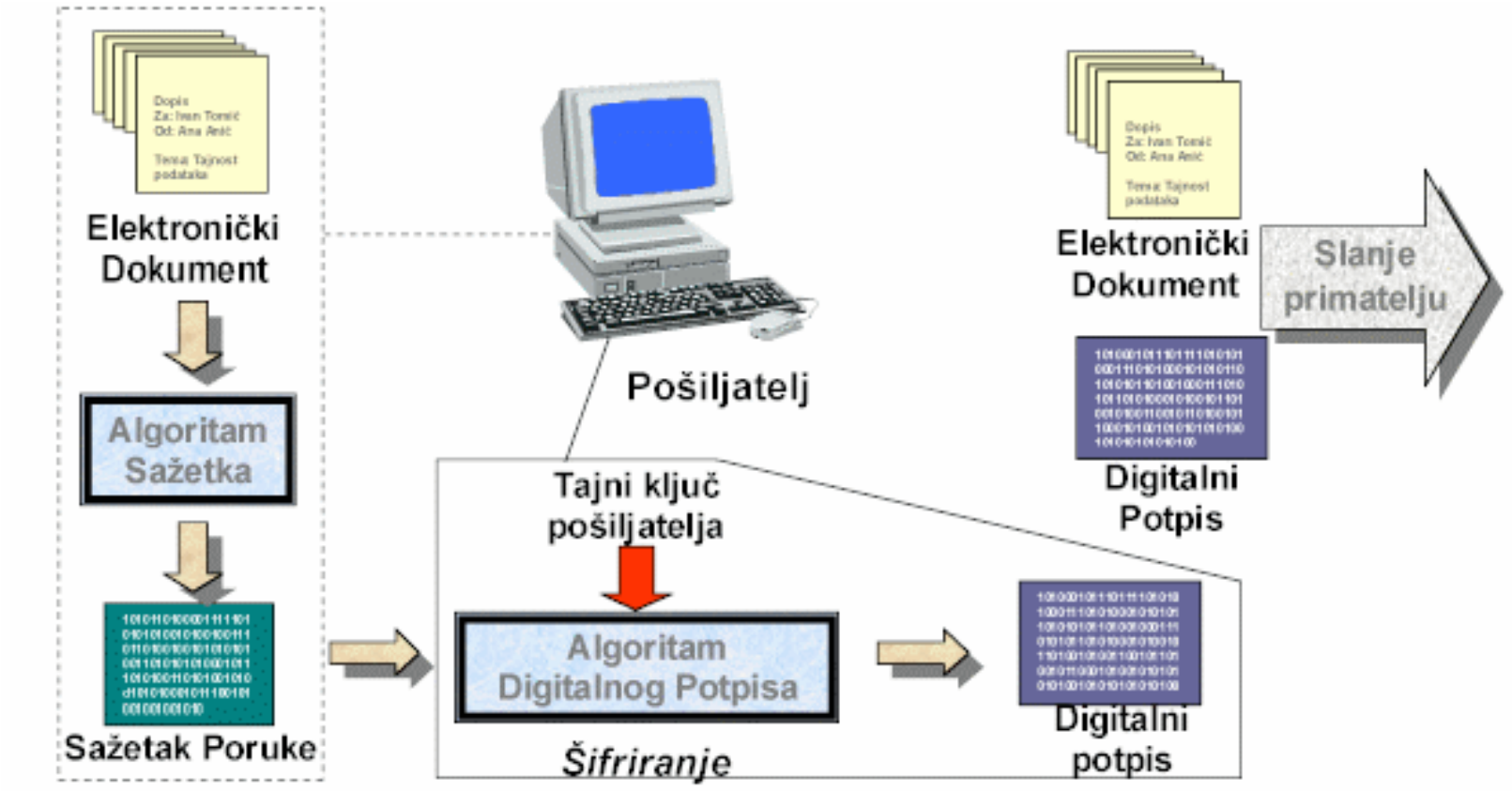
- ✓ Procesiranje poruke sastoji se od 4 slične faze koje se nazivaju “rounds”. Svaka faza sastoji se od 16 sličnih operacija baziranih na nelinearnoj funkciji F, modularnom zbrajanju i rotaciji bitova ulijevo.

$$F(X,Y,Z) = (X \wedge Y) \vee (\neg X \wedge Z)$$
$$G(X,Y,Z) = (X \wedge Z) \vee (Y \wedge \neg Z)$$
$$H(X,Y,Z) = X \oplus Y \oplus Z$$
$$I(X,Y,Z) = Y \oplus (X \vee \neg Z)$$

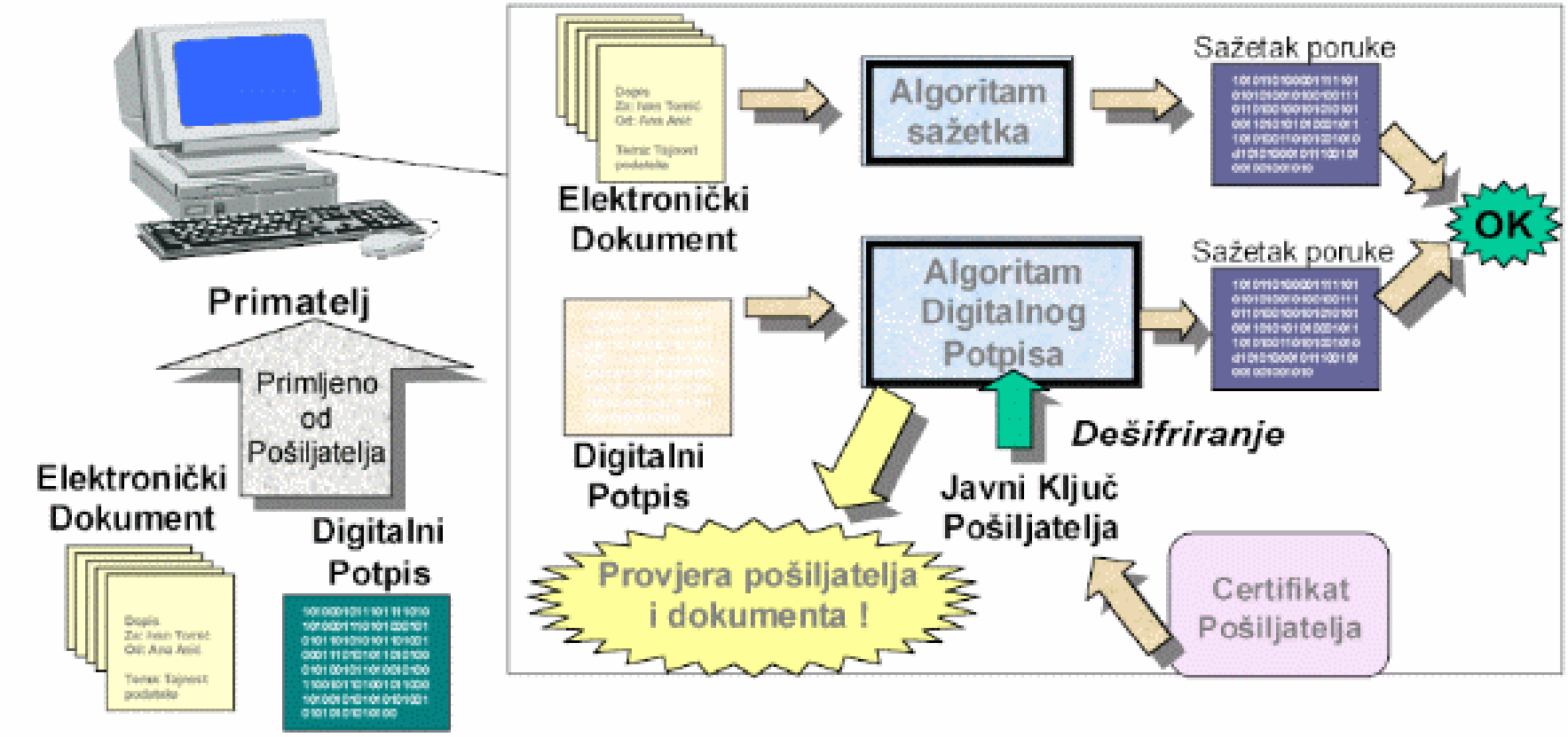
Izvod *hash* vrijednosti

- MD2, MD4, MD5 algoritmi
  - *Rivest*
  - 128-bitni izlaz
- SHA algoritam (*Secure Hash Algorithm)*
  - Dio SHS (*Secure Hash Standard*a) standarda
  - Razvile su ga i publicirale *NIST* i *NSA 1994*. godine
  - Algoritam na osnovu maksimalne duljine poruke od 2^{64} bita izračunava broj *164*-bita duljine

Hash i digitalni potpis - kreiranje



Hash i digitalni potpis - provjera



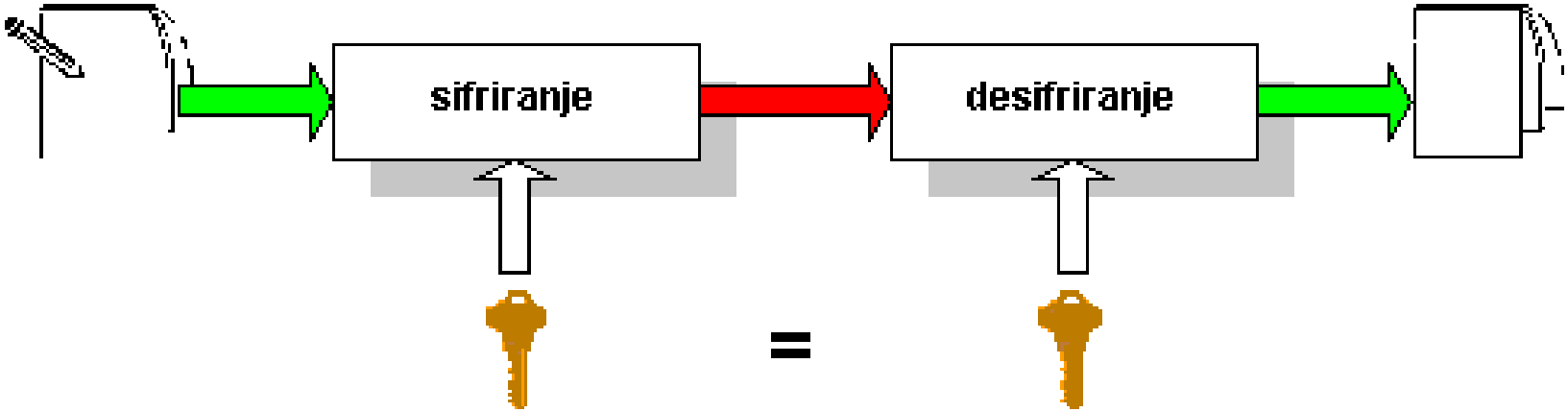
Algoritmi kriptiranja (šifriranja)

- ✓ kriptografski sustav s tajnim ključem
- ✓ kriptografski sustav s javnim ključem
  - ✓ korištenje RSA algoritma
- ✓ algoritam digitalnog potpisa (*Digital Signature Algorithm*, DSA)

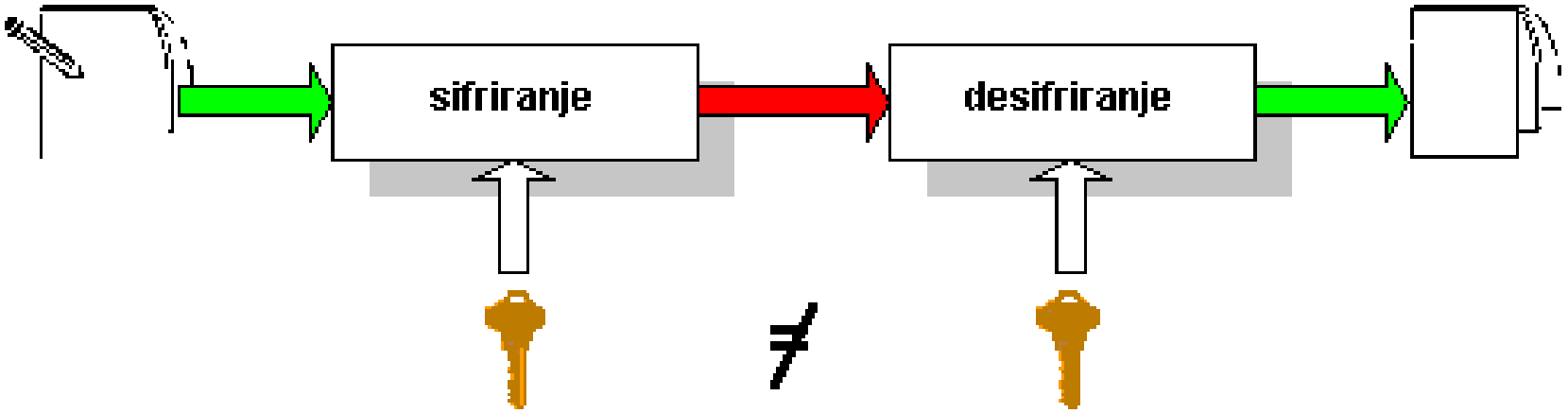
Pojmovi

- Ključ je niz alfanumeričkih znakova koji koristi kriptografski algoritam, a služi za određivanje izlaza iz funkcija kriptiranja i dekriptiranja.
- Služi za:
  - Kriptiranje (šifriranje) i dekriptiranje (dešifriranje)
  - Detekciju neovlaštenog pristupa
  - Provjeru vjerodostojnosti.
- Šifriranje je proces transformacije podataka u oblik nerazumljiv svima osim osobama koje međusobno komuniciraju.
- Dešifriranje je obratan proces, proces transformacije šifriranog teksta u korisniku prepoznatljiv oblik.

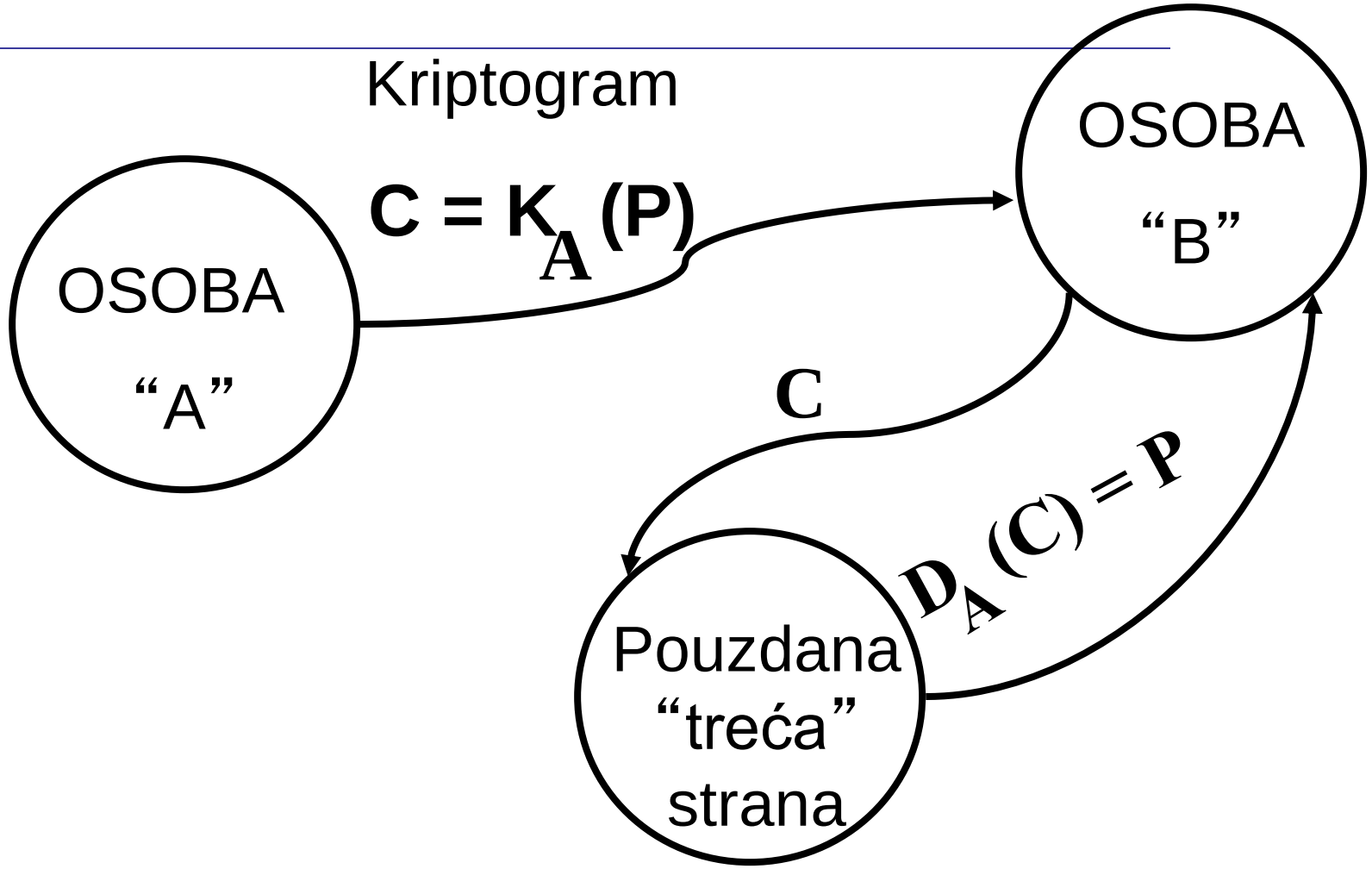
Sustav s tajnim ključem
- simetrični sustav -



Sustav s javnim ključem
- asimetrični sustav -

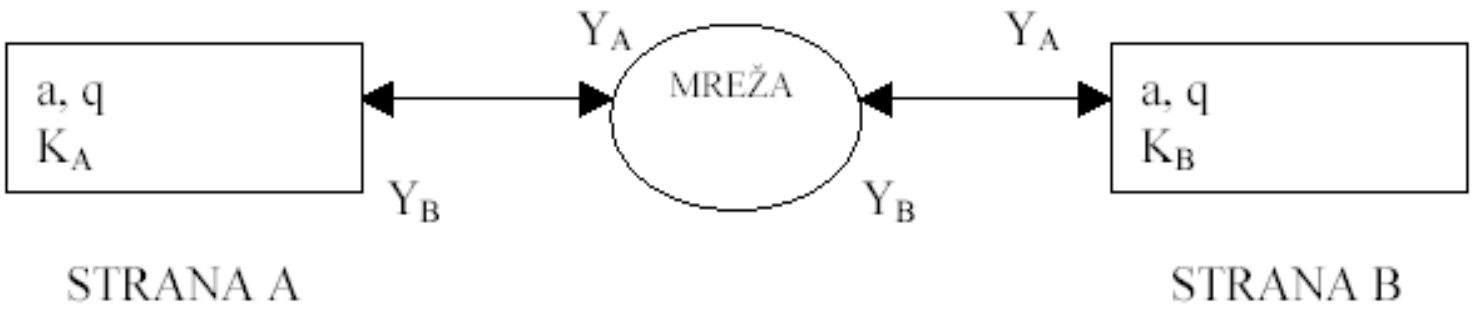


Sustav s tajnim ključem



•posjeduje transformacije šifriranja i dešifriranja osobe "A" i osobe "B"

Sustav s javnim ključem



Sustav s javnim ključem
- certifikati -

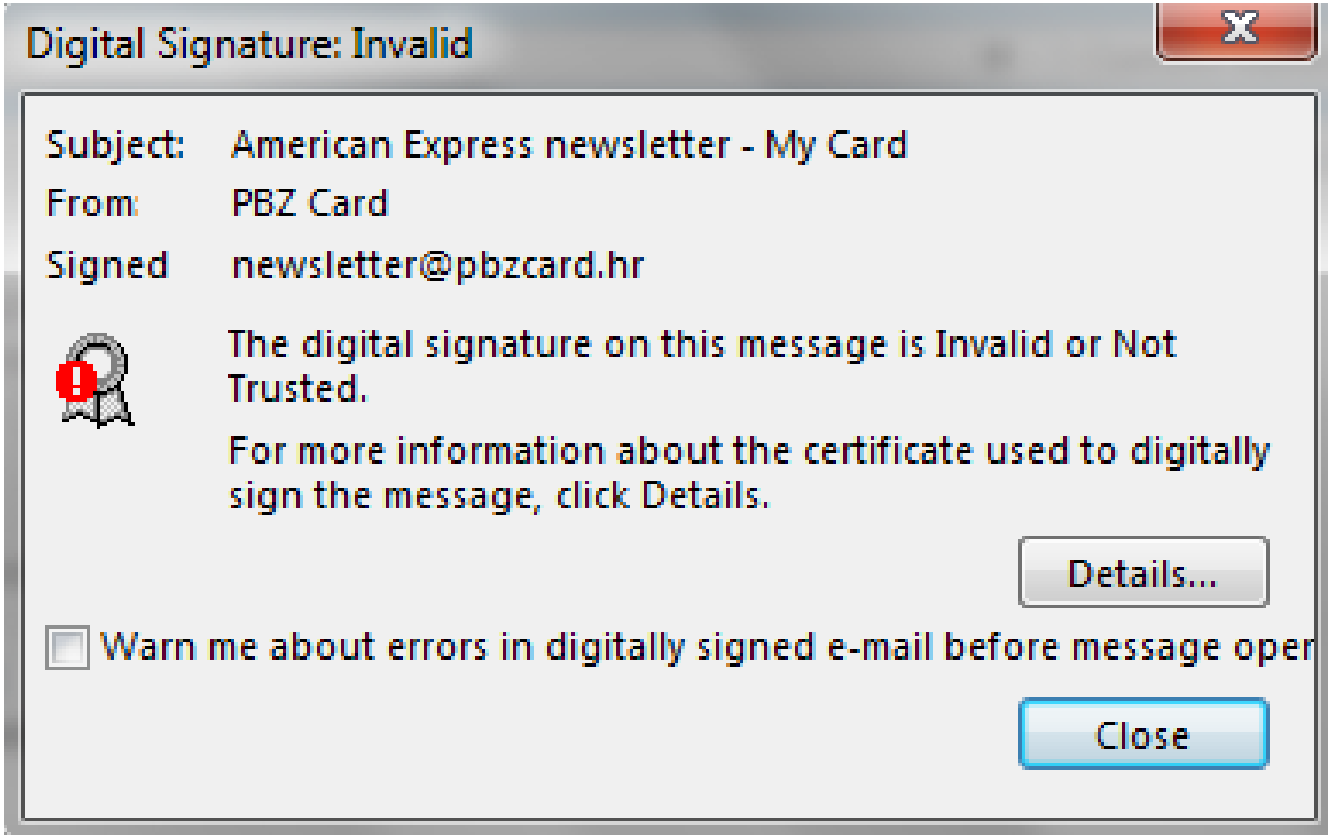
- ✓ Problem distribucije javnog ključa – kako povezati javni ključ sa vlasnikom?
- ✓ Rješenje - izdavanje certifikata.
- ✓ Certifikat - elektronički dokument koji identificira pojedinca, računalo ili neki drugi entitet koji posjeduje javni ključ.
- ✓ Treba povezati par ključeva s njihovim vlasnikom
- ✓ Izadavač certifikata ili Certificate Authority (CA)
- ✓ Publiciraju se u repozitoriju – bazi podataka u kojoj se nalaze certifikati i pripadajuće informacije

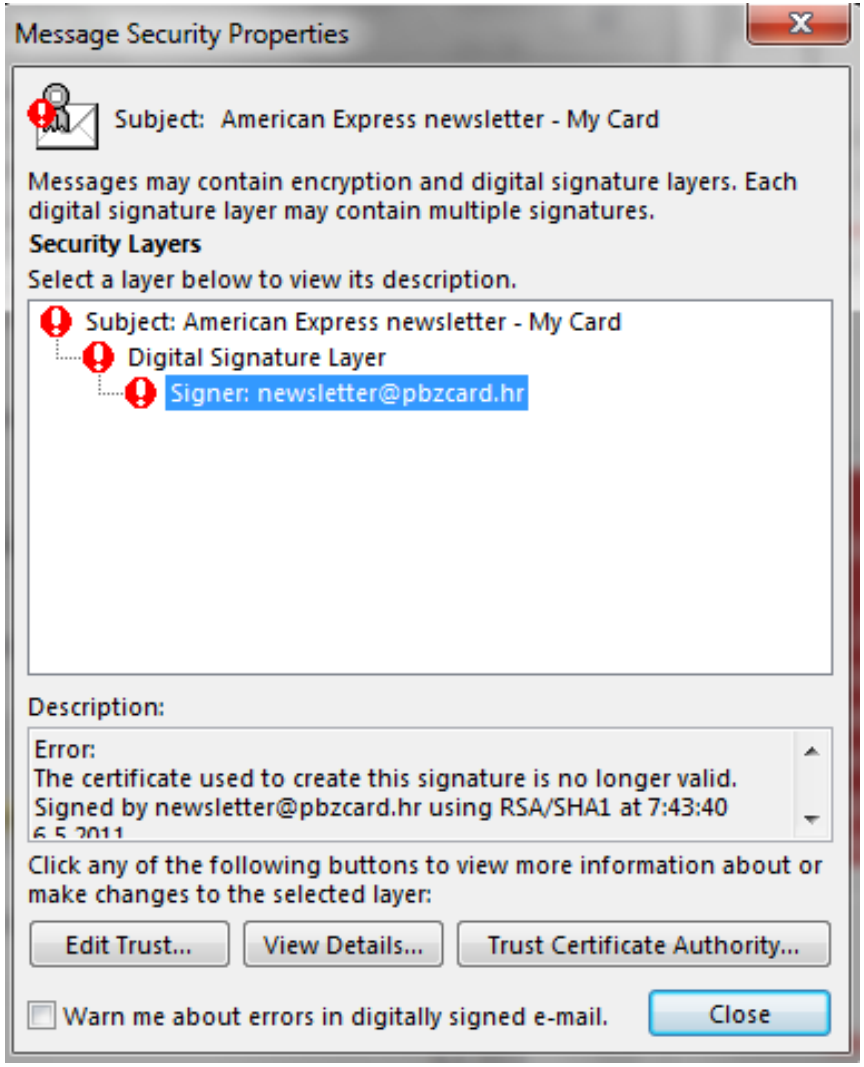
- Prema ZEP-u [NN, 2010] certifikat je „potvrda u elektroničkom obliku koja povezuje podatke za verificiranje elektroničkog potpisa s nekom osobom i potvrđuje identitet te osobe.
- Certifikat izdaje davatelj usluga izdavanja kvalificiranog certifikata, odnosno pravna ili fizička osobu koja izdaje certifikate ili daje druge usluge povezane s elektroničkim potpisima”.
- Dakle, certifikat je digitalna datoteka koju izdaju specijalizirane institucije (u RH Financijska agencija), a koja treba identificirati pojedinca, računalo ili neki drugi entitet koji posjeduje javni ključ, te povezati par ključeva s njihovim vlasnikom

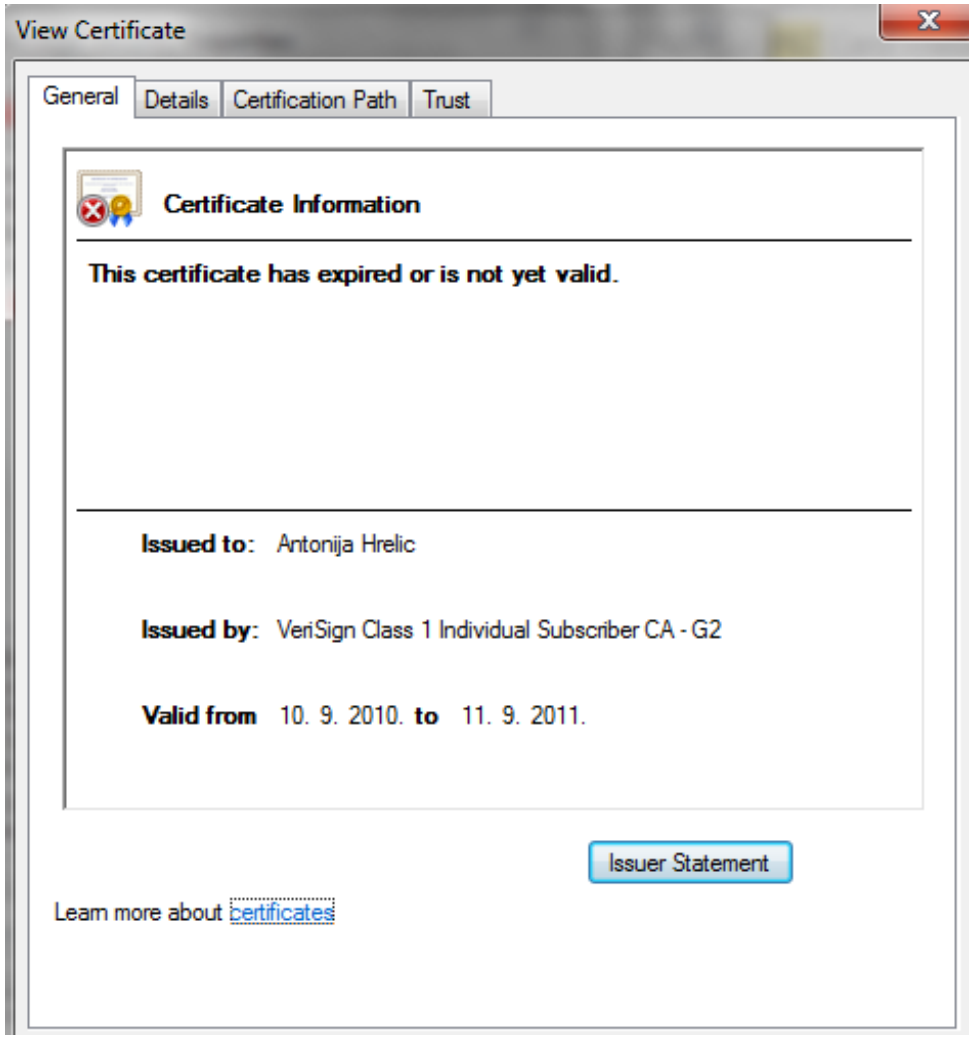
Sustav s javnim ključem
- certifikati -

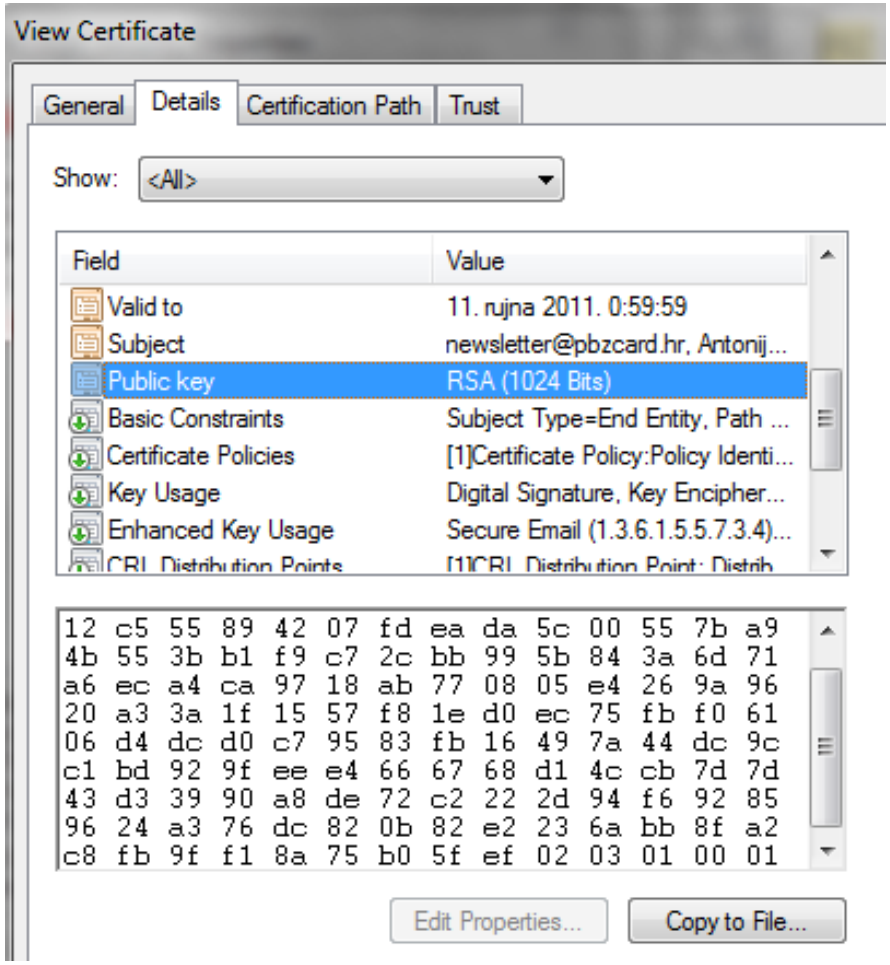
- Elementi certifikata:
  - Verzija
  - Serijski broj
  - Identifikacijska oznaka algoritma digitalnog potpisa
  - Ime ovlaštenog certifikatora (CA)
  - Vrijeme trajanja certifikata
  - Vlasnik javnog ključa

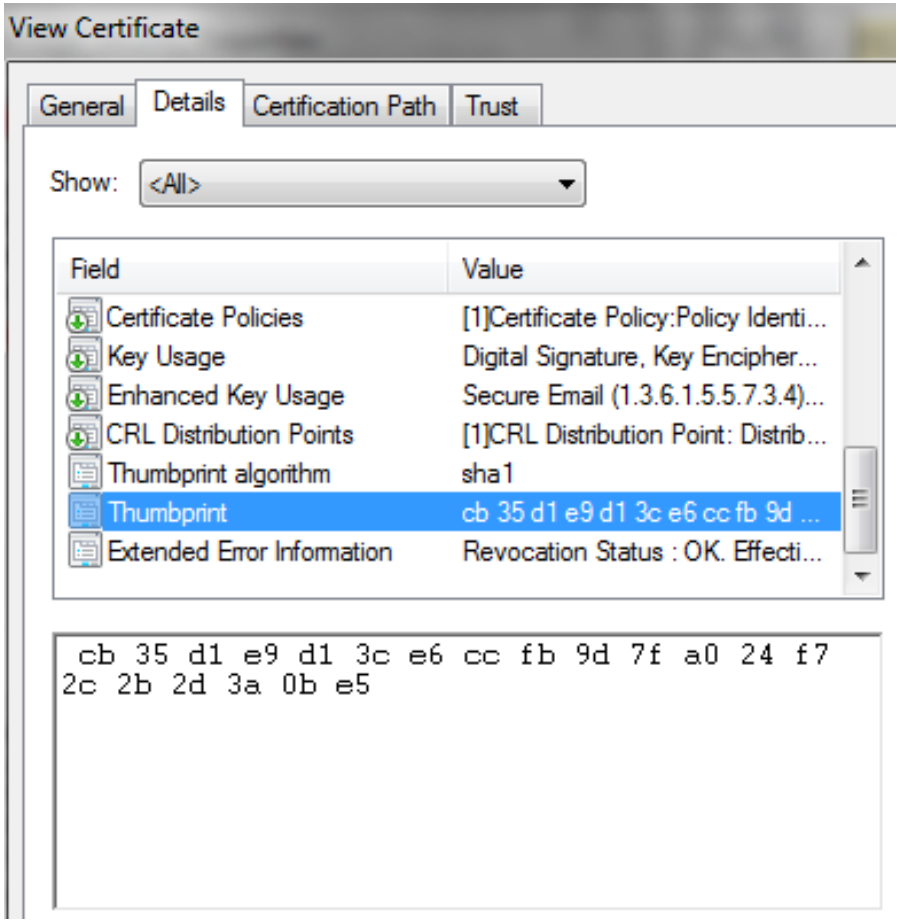

Sandro Geric
RE: Satnica CASE23
2.6.2011
Postovanje! Ne znam da li Vam je kolega Vidacic prosliejdio nasu
ante
Satnica CASE23
2.6.2011
Poštovani predavači, Program je gotov, zbornik je odštampan,
Sandro Geric
RE: Upit - rezervacija
31.5.2011
Postovana,
Lina Martinovic
RE: Upit - rezervacija
31.5.2011
Poštovani gospodine Gerić, Zahvaljujem na Vašem e-mailu te veza upit
PBZ Card
American Express Online račun za svibanj 2011
20.5.2011
American Express osobna kartica
PBZ Card
American Express newsletter - My Card
6.5.2011
<http://www.mycard.hr/> <http://www.pbzcard.hr/>
PBZ Card <newsletter@pbzcard.hr>
American Express newsletter - My Card
To Sandro Gerić
Signed By There are problems with the signature. Click the signature button for details.
If there are problems with how this message is displayed, click here to view it in a web browser.
Poštovani gospodine Gerić,
1.5.2011. započelo je novo nagradno razdoblje My Card programa! Sakupljanje prava na nagrade kod pojedinih partnera krenulo je od početka, a očekuju Vas nove, zanimljive trgovine i još više popusta i darova.
Stoga, ako ste vjeran korisnik bilo koje osobne American Express kartice sa čipom, posjetite www.mycard.hr i saznajte više!

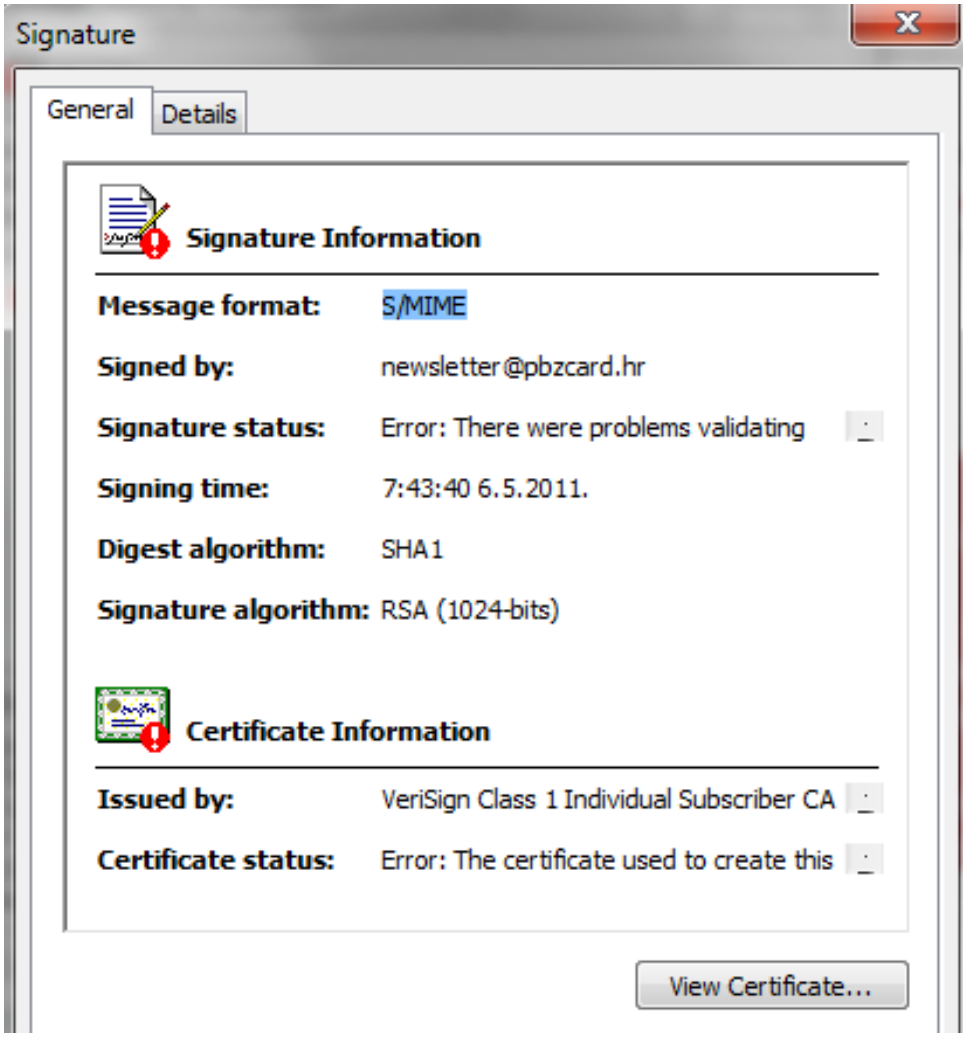

Digital Signature: Invalid
Subject: American Express newsletter - My Card
From: PBZ Card
Signed newsletter@pbzcard.hr
The digital signature on this message is Invalid or Not Trusted.
For more information about the certificate used to digitally sign the message, click Details.
Details...
Warn me about errors in digitally signed e-mail before message oper
Close

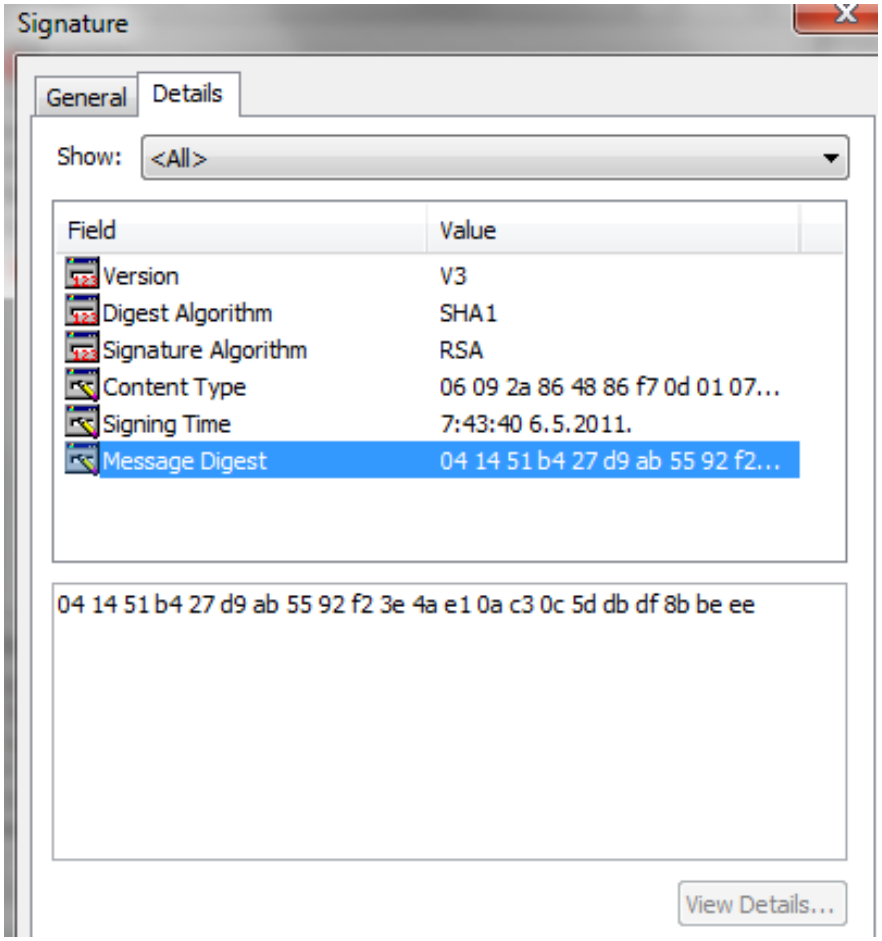

Message Security Properties
Subject: American Express newsletter - My Card
Messages may contain encryption and digital signature layers. Each digital signature layer may contain multiple signatures.
Security Layers
Select a layer below to view its description.
Subject: American Express newsletter - My Card
Digital Signature Layer
Signer: newsletter@pbzcard.hr
Description:
Error:
The certificate used to create this signature is no longer valid.
Signed by newsletter@pbzcard.hr using RSA/SHA1 at 7:43:40
6.5.2011
Click any of the following buttons to view more information about or make changes to the selected layer:
Edit Trust...
View Details...
Trust Certificate Authority...
Warn me about errors in digitally signed e-mail.
Close


View Certificate
General
Details
Certification Path
Trust
Certificate Information
This certificate has expired or is not yet valid.
Issued to: Antonija Hrelic
Issued by: VeriSign Class 1 Individual Subscriber CA - G2
Valid from 10. 9. 2010. to 11. 9. 2011.
Issuer Statement
Learn more about certificates


View Certificate
General
Details
Certification Path
Trust
Show:
<All>
Field
Value
Valid to
11. rujna 2011. 0:59:59
Subject
newsletter@pbzcard.hr, Antonij...
Public key
RSA (1024 Bits)
Basic Constraints
Subject Type=End Entity, Path ...
Certificate Policies
[1]Certificate Policy:Policy Identi...
Key Usage
Digital Signature, Key Encipher...
Enhanced Key Usage
Secure Email (1.3.6.1.5.5.7.3.4)...
12 c5 55 89 42 07 fd ea da 5c 00 55 7b a9
4b 55 3b b1 f9 c7 2c bb 99 5b 84 3a 6d 71
a6 ec a4 ca 97 18 ab 77 08 05 e4 26 9a 96
20 a3 3a 1f 15 57 f8 1e d0 ec 75 fb f0 61
06 d4 dc d0 c7 95 83 fb 16 49 7a 44 dc 9c
c1 bd 92 9f ee e4 66 67 68 d1 4c cb 7d 7d
43 d3 39 90 a8 de 72 c2 22 2d 94 f6 92 85
96 24 a3 76 dc 82 0b 82 e2 23 6a bb 8f a2
c8 fb 9f f1 8a 75 b0 5f ef 02 03 01 00 01
Edit Properties...
Copy to File...


View Certificate
General
Details
Certification Path
Trust
Show:
<All>
Field
Value
Certificate Policies
[1]Certificate Policy:Policy Identi...
Key Usage
Digital Signature, Key Encipher...
Enhanced Key Usage
Secure Email (1.3.6.1.5.5.7.3.4)...
CRL Distribution Points
[1]CRL Distribution Point: Distrib...
Thumbprint algorithm
sha1
Thumbprint
cb 35 d1 e9 d1 3c e6 cc fb 9d ...
Extended Error Information
Revocation Status : OK. Effecti...
cb 35 d1 e9 d1 3c e6 cc fb 9d 7f a0 24 f7
2c 2b 2d 3a 0b e5


Signature
General
Details
Signature Information
Message format: S/MIME
Signed by: newsletter@pbzcard.hr
Signature status: Error: There were problems validating
Signing time: 7:43:40 6.5.2011.
Digest algorithm: SHA1
Signature algorithm: RSA (1024-bits)
Certificate Information
Issued by: VeriSign Class 1 Individual Subscriber CA
Certificate status: Error: The certificate used to create this
View Certificate...


Signature
General
Details
Show:
<All>
Field
Value
Version
V3
Digest Algorithm
SHA1
Signature Algorithm
RSA
Content Type
06 09 2a 86 48 86 f7 0d 01 07...
Signing Time
7:43:40 6.5.2011.
Message Digest
04 14 51 b4 27 d9 ab 55 92 f2...
04 14 51 b4 27 d9 ab 55 92 f2 3e 4a e1 0a c3 0c 5d db df 8b be ee
View Details...

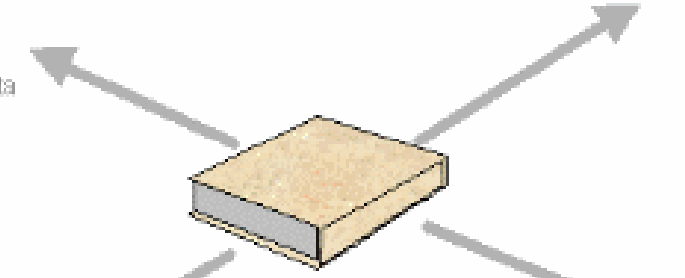
Sustav s javnim ključem
- certifikati -

Certifikator (CA)

- Kreira i opoziva certifikate
- Objavljuje listu aktualnih i opozvanih certifikata

Registar Certifikata

- Sadrži aktualne i opozvane certifikate



Registrator (RA)

- Provjerava i jamči identitet korisnika
- Odobrava zahtjeve za izdavanje certifikata

Pošiljatelj

- Dobiva Certifikat od CA
- Koristi tajni ključ za izradu digitalnog potpisa

Sustav s javnim ključem – RSA algoritam (Rivest, Shamir, Adleman)

- *RSA* sustav se temelji na teoriji brojeva, a princip njegova djelovanja je slijedeći:
  - odabiru se dva velika prosta (prim) broja *P* i *Q* (P>10100 i Q>10100);
  - odredi se umnožak *N=P*Q*, te vrijednost *(P-1)*(Q-1)* koju označimo sa *L(N)*;
  - odabire se broj *d*, tako da bude *max(P,Q)<d<L(N)*;
  - izračunava se broj *e* tako da bude *0<e<L(N)* i da je *(e*d)modL(N)=1*, što je isto kao da odredimo najmanji *k* za koji vrijedi *e*d=k*L(N)+1*;
  - par *(e,N)* se proglašava javnim ključem.
  - Iz teorije brojeva je poznato da uz *0≤M<N*, za tako izračunate *e* i *d* vrijedni da je: *(MemodN)dmodN=MedmodN=MkL(N)+1modN=M*

Sustav s javnim ključem – RSA algoritam

- Kriptosustav s javnim ključem djeluje na slijedeći način:
  - kriptiranje se vrši tajnim ključem *(e,N)* tako da se:
    - razgovjetni tekst kodira u niz cijelih brojeva *Mi* koji smiju poprimiti vrijednosti iz intervala od *0* do *N-1*,
    - svaki od tih cijelih brojeva se kriptira u *Ci=(Mi)emodN*.
  - dekriptiranje se obavlja na slijedeći način:
    - *Ci* se dekriptira u razgovjetni oblik *Mi=(Ci)dmodN*;
    - niz brojeva *Mi* se dekodira u izvorni tekst.

Sustav s javnim ključem – RSA algoritam (Rivest, Shamir, Adleman)

Na strani pošiljatelja:

P - dokument

DP - digitalni potpis

$$DP = P^{d} \ (\mathrm{mod}\ n)$$

pomoću privatnog ključa (d, n)

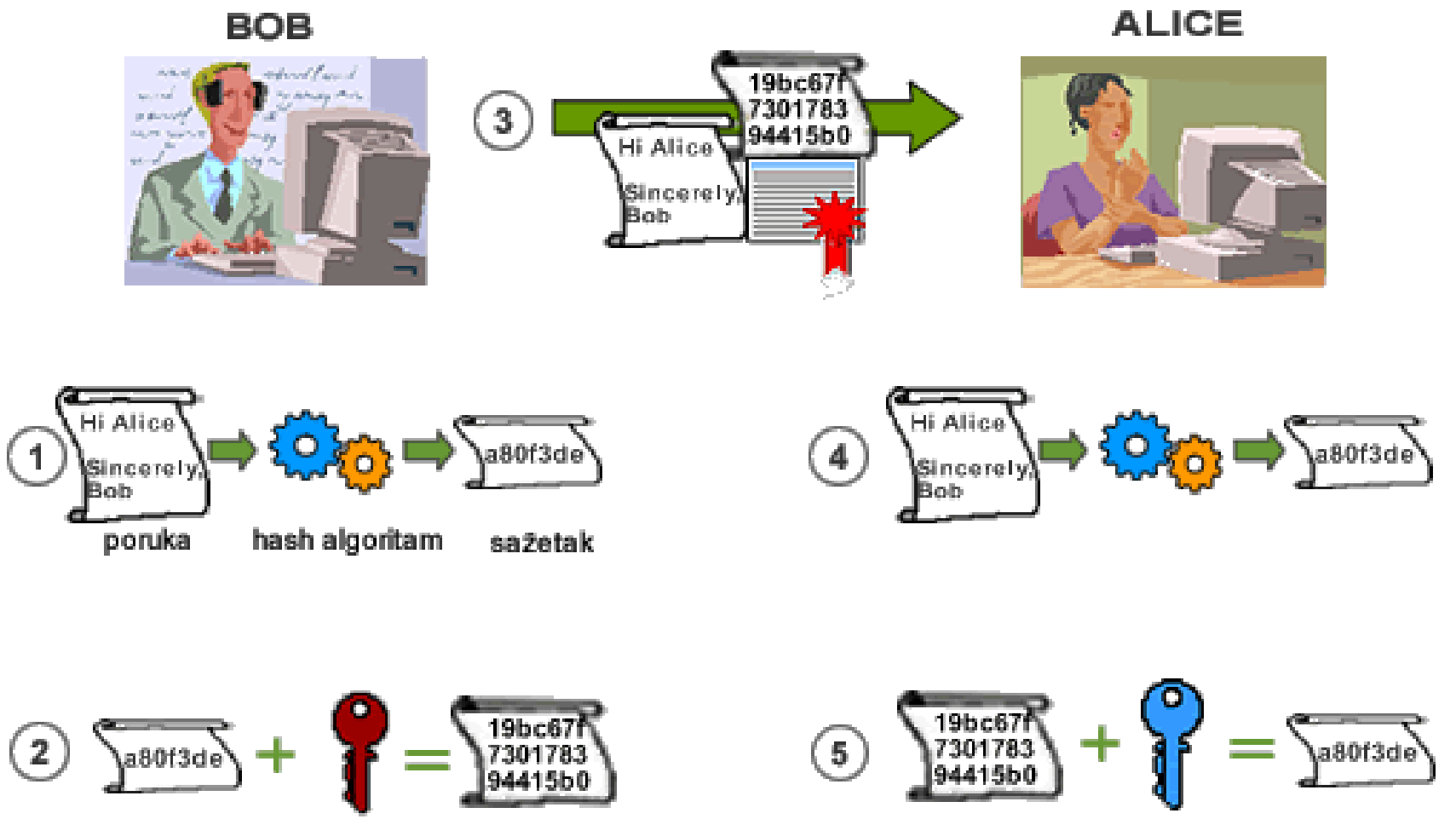
Na strani primatelja:

$$P = DP^{e} \ (\mathrm{mod}\ n)$$

pomoću javnog ključa (e, n)
izračunava P

provjerava $P = P$

Sustav s javnim ključem – RSA algoritam



Sustav s javnim ključem – RSA algoritam

1. Hash funkcijom Bob računa sažetak poruke koju šalje Alici,
2. Bob kriptira svojim tajnim ključem sažetak poruke i na taj način kreira digitalni potpis,
3. Zajedno s originalnim dokumentom, Bob šalje i digitalni potpis,
4. Alice dobiva Bobovu potpisanu poruku, a iz originalne poruke izračuna sažetak,
5. Alice dekriptira digitalni potpis Bobovim javnim ključem, te uspoređuje dekriptirani sažetak s onim koji je sama izračunala. Ako su jednaki, Alice je sigurna da je Bob poslao poruku i da se poruka nije mijenjala tokom slanja (integritet poruke). Bob ne može poreći da je on poslao poruku, jer se digitalni potpis može dekriptirati samo njegovim javim ključem, a kriptirati njegovim tajnim ključem.

DSA algoritam

- **D**igital **S**ignature **A**lghorithm
- Definira proces kreiranja (generiranja) i provjere (verifikacije) digitalnog potpisa
- Razvijen od strane *National Security Agency – NSA*, a *National Institute of Standards and Technology – NIST* ga je standardizirao unutar posebnog standarda za digitalni potpis (*Digital Signature Standard – DSS)*

DSA algoritam

- ✓ Sigurnost DSA temelji se na problemu izračunavanja diskretnog algoritma
- ✓ Koristi se ključ veličine 1024 bita
- ✓ Primjenjuje se na *hash* vrijednost, a ne na cijeli dokument

XML digitalni potpis

- XML digitalni potpis karakterizira svojstvo da omogućuje „potpisivanje“ samo određenog dijela dokumenta, a ne samo čitavog dokumenta kao što je to slučaj sa “običnim” i “naprednim” e-potpisom. Postoje 3 osnovne vrste XML digitalnog potpisa:
  - *detached* - potpisani podaci i XML potpis su odvojeni;
  - *enveloping* - potpisani podaci su ugrađeni u XML potpis i
  - *enveloped* – potpis je ugrađen u podatke koje potpisuje.

XML digitalni potpis

- Osnovni element XML digitalnog potpisa definiran pripadajućom XML Schemom jest *Signature* element koji ima sljedeću strukturu [W3C, 2010]:

```
<Signature ID?>
      <SignedInfo>
        <CanonicalizationMethod/>
        <SignatureMethod/>
        (<Reference URI? >
          (<Transforms>)?
          <DigestMethod>
          <DigestValue>
        </Reference>)+
      </SignedInfo>
      <SignatureValue>
     (<KeyInfo>)?
     (<Object ID?>)*
   </Signature>
```

XML digitalni potpis

- *Signature ID* je korijenski element definicije digitalnog potpisa unutar kojeg se nalazi element *SignedInfo* kojim se definira primijenjeni kanonizacijski algoritam te algoritam korišten za sam potpis
- Elementi *CanonicalizationMethod* i *SignatureMethod* služe za zapis naziva korištenih algoritama.
- Uz prethodno navedena tri osnovna elementa specifikacije XML digitalnog potpisa, svaki resurs na koji se primjenjuje digitalni potpis karakteriziran je *Reference* elementom, to jest s URI (*Uniform Resource Identifier*) atributom. Elementom *Transform* definira se sadržaj koji je korišten za provođenje niza transformacija kod XML digitalnog potpisivanja, a element *Algorithm* definira naziv algoritma koji se pri tome koristi.
- *DigestValue* element služi za pohranu kreiranog sažetka u okviru XML digitalnog potpisa, najčešće kodiranog base64 algoritmom, a element *SignatureValue* služi za zapis stvarne vrijednosti digitalnog potpisa.
- Poveznice na primijenjene ključeve, certifikate i slične informacije potrebne kod digitalnog potpisivanja nalaze se u opcionalnom *KeyInfo* elementu[W3C, 2010].

XML digitalni potpis

- XML digitalno potpisivanje provodi se na sljedeći način:
  - određivanje sadržaja koji se potpisuje.
  - određivanje sažetka za sadržaje koji se žele potpisati. Adrese sadržaja koji se potpisuju definiraju se pomoću *Reference* elementa, a izračunata vrijednost se pohranjuje u *DigestValue* element.
  - Za potpisane elemente izračunava se vrijednost sažetka *SignedInfo* elementa koja se zapisuje u *SignatureValue* element.
  - Sve poveznice na primijenjene ključeve, certifikate i slične informacije potrebne kod digitalnog potpisivanja nalaze se u opcionalnom *KeyInfo* elementu. Najčešće se radi o poveznici i informacijama vezanim uz X.509 certifikate.
  - Posljednji korak kod XML digitalnog potpisivanja jest objedinjavanje svih elemenata unutar *Signature* elementa.
  - PRIMJER

XML digitalni potpis

- Provjera XML digitalnog potpisa se svodi na provjeru *SignedInfo* elementa.
  - Sastoji se ponovnom izračunavanju njegove vrijednost, pri čemu se koriste isti algoritmi specificirani potpisom.
  - Upotrebom javnog ključa (koji se iščita iz informacija certifikata – KeyInfo element) vrši provjera da li je vrijednost *SignatureValue* elementa ispravna za sažetak *SignedInfo* elementa, te se vrijednost sažetka *SignedInfo* elementa uspoređuje sa vrijednošću unutar svakog *Reference* elementa koji se odnosi na *DigestValue*.
  - Ako su uspoređene vrijednosti jednake nije došlo do promjene u potpisanim sadržajima i sigurni smo da su sadržaji došli od osobe koja je vlasnik potpisa [W3C, 2010].

Štićeni prijenos podataka i digitalni potpis

- ✓ S/HTTP
- ✓ S/MIME
- ✓ SSL
- ✓ PGP

Štićeni prijenos podataka
- S/HTTP -

- ✓ *Secure Hypertext Transfer Protocol*
- ✓ Nadopuna standardnom *HTTP*-u (*Hypertext Transfer Protokolu*) s ciljem dodavanja sigurnih servisa putem kriptografije.
- ✓ Temelji se na početnom pregovaranju između korisnika i poslužioca oko vrste kriptografije kojom će komunicirati.

Štićeni prijenos podataka
- S/HTTP -

- četiri načina razmjene ključeva:
  - RSA - razmjenjuju se standardni javni *RSA* ključevi
  - *in-band* - transport ključeva putem *S/HTTP* zaštićene poruke
  - *out-band* - vanjski dogovor razmjene ključeva
  - *Kerberos* - ključ se dobavlja sa *Kerberos* poslužioca

Štićeni prijenos podataka
- S/MIME -

- ✓ *Secure Multi-purpose Internet Mail Extensions*
- ✓ Internet standard za kodiranje i potpis poruka za siguran prijenos preko Interneta.
- ✓ S/MIME poruke mogu biti sastavljene od više dijelova s kombinacijama teksta, glasa, i grafike.

Štićeni prijenos podataka
- S/MIME -

- ✓ nadogradnja na postojeći Internet *MIME* protokol s mogućnošću digitalnog potpisivanja i kriptiranja elektronske pošte
- ✓ *S/MIME* standard je prihvatio velik broj proizvođača softvera (*ConnectSoft, Frontier, FTP Software, Microsoft, Lotus, Banyan, NCD, SecureWare, VeriSign, Netscape* i *Novell*)

Štićeni prijenos podataka
- SSL -

- *Secure Socket Layer*
- *protokol rukovanja* (*Handshake Protocol*)
- podržava identificiranje te osiguravanje autentičnosti i poslužioca i korisnika
- transparentan protokol s mogućnost da se drugi protokoli (*HTTP*, *FTP*, *Telnet* i *Rsh)* na jednostavan način nadograde kao komunikacijski sloj
- Sam protokol rukovanja se sastoji od dvije faze:
  - identifikacije poslužioca i
  - identifikacije korisnika.

Štićeni prijenos podataka
- PGP -

- *Pretty Good Privacy*
- De facto standard za šifriranje poruka elektronske pošte.
- Ostvarenje tajnosti, autentičnosti i integriteta poruke elektroničke pošte.
- Princip rada:
  - Generira se javni i privatni ključ,
  - Korisnik unosi tajni izraza *(pass phrase),*
  - Generirani ključevi se pohranjuju u certifikatima ključa,
  - Objavljuje se javni ključ.

Prednosti i nedostaci digitalnog potpisa

\+

- ✓ nemogućnost prevare
- ✓ integritet poruka
- ✓ pravni zahtjevi
- ✓ otvoreni sustavi

\-

- ✓ troškovi

Slijepi potpis

- ✓ Oblik digitalnog potpisa.
- ✓ Razvijen je zbog potrebe da se osigura autentičnost podataka, uz istovremeno osiguranje anonimnosti osobe koja je potpisala takav podatak.
- ✓ Važna primjena u sferi e-plaćanja.
- ✓ Matematička podloga – jednosmjerne funkcije (matematičke funkcije čije vrijednosti je moguće jednostavno odrediti, a vrlo je teško izračunati njihove inverzne vrijednosti).

Slijepi potpis

- Više modela:
  - *Random oracle model*
  - *Compexity-based proofs*
- Problem koji se javlja kod slijepog potpisa je neostavljanje tragova – slijepi potpis predstavlja priliku za “savršen” zločin.
- “Pravedni” slijepi potpis.

Slijepi potpis

- ✓ **“Savršen zločin”**
- ✓ otmičar zatraži plaćanje otkupnine u e-kovanicama
- ✓ otmičar objavi u novinama skup markiranih stringova (skrivenih svojim potpisom) - npr. 20 je skriveno množenjem s 5
- ✓ banka potpiše string 100 (npr. množenjem s 10) i objavi potpisane stringove u novinama
- ✓ otmičar kupi novine i skine maskirajući faktor (podijeli s 5) i ima potpisane e-kovanice

Slijepi potpis

- Princip rada:
  - Osoba A želi da osoba B potpiše poruku M
    - A množi M s proizvoljnim brojem ili *maskirajućim faktorom (MF)*
    - Maskiranu poruku (M*MF) osoba A šalje osobi B
    - Osoba B ne može pročitati poruku M (jer je maskirana), pa B ne zna sadržaj poruke M
    - Osoba B potpisuje maskiranu poruku M*MF sa svojim privatnim ključem, i vraća takvu poruku k osobi A
    - Osoba A dijeli primljenu poruku s maskirajućim faktorom (MF) što rezultira s originalnom porukom koja je potpisana od strane B (npr. banke)

Slijepi potpis

- Problem – kako potpisati dokument (npr. apoen e-novca) koji ne možete pročitati?
  - Rješenje: više potpisa za različite apoene.
- Kovanje e-kovanica:
  - A šalje nepotpisane, maskirane e-kovanice u banku, uz oznaku ser. broja kovanica i apoena
  - Banka ih potpisuje slijepim potpisom i s računa osobe A skida novčani iznos
  - Banka zna vlasnika e-kovanica, ali ne zna njihove serijske brojeve – ANONIMNOST
  - Banka šalje potpisane e-kovanice osobi A koja ih demaskira.

Slijepi potpis

- Trošenje e-kovanica:
  - Osoba A plaća osobi C
  - C provjerava da li kovanice već nisu potrošene i polaže kovanice banku izdavača
  - Banka plača prodavaču (C) novcem.
- Problem *replay-a* – kako se osigurati u slučaju višestrukog trošenja istih e-kovanica (problem kod off-line rada).
  - Kupac je anoniman – tko je odgovoran ako banka dobije već potrošene novčanice?

Slijepi potpis

- Modifikacija:
  - *Chaum double spending protocol*
  - A želi 100 e-kovanica
  - A šalje 200 e-kovanica banci na potpis
    - Za svaku e-kovanicu A kombinira “b” različitih slučajnih brojeva s brojem vlastitog računa i serijskim brojem kovanice (ex ILI funkcija), i maskira e-kovanice
  - Banka odabire 100 od 200 e-kovanica, potpisuje ih i šalje k A, te od A traži slučajne brojeve za preostalih 100 e-kovanica (da bi pročitala broj računa)
    - Da li je A dao ispravan broj računa?
      - Da, jer je banka odabirala e-kovanice za potpis.

Slijepi potpis

- Modifikacija:
  - A plača prodavaču C s e-kovanicama
  - Zajedno s kovanicama C zaprima i broj računa od A (u XOR obliku sa slučajnim brojem)
  - Ako je došlo do *replaya* onda je banka dobila za isti serijski broj kovanice dva različita broja koja su zaprimili prodavači C i C’
  - Njihovom i kombinacijom slučajnog broja “b” te primjenom XOR funkcije banka može doći do originalnog broja računa osobe A i identificirati počinitelja replaya.

Slijepi potpis

- ✓ Dobiven je dobar sustav:
  - ✓ Ako nije došlo do *replaya* osigurana je anonimnost kupca.
  - ✓ Ako je došlo do *replaya* moguće je identificirati počinitelja.

Slijepi potpis

- Ukoliko je broj računa 12, što je hex 0C= 00001100
- A odabire serijski broj 100 i maskirajući broj 5
- Zahtijeva od banke kovanicu sa serijskim brojem 100 x 5 = 500
- A odabire slučajan broj b i kreira b slučajnih brojeva za tu kovanicu. Uzmimo b = 6
- A radi XOR svakog slučajnog broja sa vlastitim brojem računa

i	rač.	sluč.	rač. XOR sluč.
0	0C	1B	17
1	0C	13	1F
2	0C	09	05
3	0C	05	09
4	0C	2B	27
5	0C	11	1D

Slijepi potpis

- Prodavac B zaprima kovanice od A. Pronalazi b i odabire slučajan broj sa b bitova, npr. 111010
- Za svaku poziciju bita u kojoj prodavacev B-ov broj ima 1, on zaprima slučajan broj od A za tu poziciju
- Za svaku poziciju sa 0, prodavac B zaprima broj računa od A XOR sa slučajnim brojem A za tu poziciju

i	rač.	sluč.	rač XOR sluč	B bit	B zaprima
0	0D	1B	17	0	17
1	0D	13	1F	1	13
2	0D	09	05	0	05
3	0D	05	09	1	05
4	0D	2B	27	1	2B
5	0D	11	1D	1	11

- Prodavac B šalje zadnji stupac u banku kada polaže kovanicu

Slijepi potpis

- Sada A pokušava potrošiti kovanicu ponovno kod prodavaca C. On pronalazi b=6 i odabire slučajni broj 010000.
- Prodavac C prolazi istu proceduru kao B i šalje brojeve koje je primio u banku kamo polaže kovanicu

i	rač.	sluč.	rač. XOR sluč.	C-ov bit	C zaprima
0	0D	1B	17	0	17
1	0D	13	1F	0	13 1F
2	0D	09	05	0	05
3	0D	05	09	0	09
4	0D	2B	27	1	2B
5	0D	11	1D	0	1D

Slijepi potpis

- Banka odbija platiti C-u, jer je kovanicu položio B.
- Banka kombinira podatke od B i C korištenjem XOR gdje je pronašla podatke iz dva izvora

i	rač.	sluč.	rač. XOR sluč.	sluč. XOR rač. XOR sluč.	Vlasnik rač.
0	0C		17		
1	0C	13	1F	0C	A
2	0C		05		
3	0C	05	09	0C	A
4	0C	2B			
5	0C	11	1D	0C	A

- Ovo identificira A kao varalicu. Niti B niti A niti banka nisu to mogli napraviti sami.

Najčešći sigurnosni napadi u e-poslovanju

Prisluškivanje žičnih i bežičnih komunikacija

- iskorištava se nedovoljna (ili često puta i nepostojanje) zaštita žičnih vodova koje informacijski i telekomunikacijski sustav koristi za komunikaciju između svojih dijelova, a s ciljem prisluškivanja linija kojima putuju podaci, te nelegalnog pristupa tim istim podacima

Uskraćivanje ili degradacija usluge *Distributed Denial of Service* (*DDoS*)

- Prijetnja koja djeluje na smanjivanje dostupnosti informacijskog resursa.
- Dostupnost, u kontekstu sigurnosti IS-a, podrazumijeva "da sve unaprijed definirane računalne mogućnosti, sama računalna infrastruktura, programska podrška i mogućnosti upotrebe moraju biti na raspolaganju legalnom korisniku".
- Do uskraćivanja usluga može doći na različite načine, npr. fizičkim napadom (palež, eksplozije, fizičko uništenje opreme), isključivanjem pomoćnih uređaja (el. energija, generati, klimatizacijski uređaji), prirodnim katastrofama i slično, a svi oni rezultiraju fizičkim oštećenjem opreme ili zatrpavanje i preopterećenjem računalnog sustava (brisanje podataka, pretrpavanje prostora na disku, preopterećivanje procesora računala, zagušivanje mrežnog prometa)

Stražnja vrata

- stražnja vrata – *back door*
- programsko rješenje koje se koristi kod razvoja softvera kako bi se olakšao pristup programera pojedinim dijelovima aplikacije može biti zloupotrebljeno za privilegiran pristup resursima računala bez ikakvih sigurnosnih provjera i kontrola

Otmice sjednica

- otmice sjednica – *hijacking*
- korištenje tuđeg računala za izvršavanje nelegalnih aktivnosti u vremenu kad ga korisnik računala ne koristi, a nije se odjavio sa sustava (npr. prilikom odlaska na kratku pauzu ostavlja se nezaštićeno računalo);

Vremenski napadi

- vremenski napadi – sofisticiran oblik sigurnosnih prijetnji koji iskorištava princip rada računala i redoslijed izvršavanja naredbi i procesa.
- Cilj je promjenom prioriteta, odnosno redoslijeda izvršavanja procesa na računalu zaobići sigurnosne mjere i iskoristiti računalne resurse za neovlaštene aktivnosti.

Malware programi

- maliciozni računalni programi – programi poput crva, logičkih bombi, trojanskih konjeva, zamki, virusa, *hoaxeva*
- zlonamjerni te da se bez privole korisnika instaliraju na njegovo računalo (*malware* programi)
- mogućnost brzog, neprimjetnog i učinkovitog širenja, da zaobilazi obrambene mehanizme računala, da može preživjeti u zaraženom računalu bez da bude otkriven i uništen, te da na različite načine iskorištava zaraženo računalo i njegove resurse

Spoofing

- *spoofing* – "je napad u kojemu počinitelji dolaze do željenih podataka koristeći se ponajprije slabostima Interneta i nedovoljnom pažnjom korisnika".
- Riječ je o napadu u kojem počinitelji od korisnika pomoću prijevara izvlače povjerljive podatke (poput brojeva kreditnih kartica, IP adresa, zaporki, e-mail adresa) te ih iskorištavaju za kasnije aktivnosti.

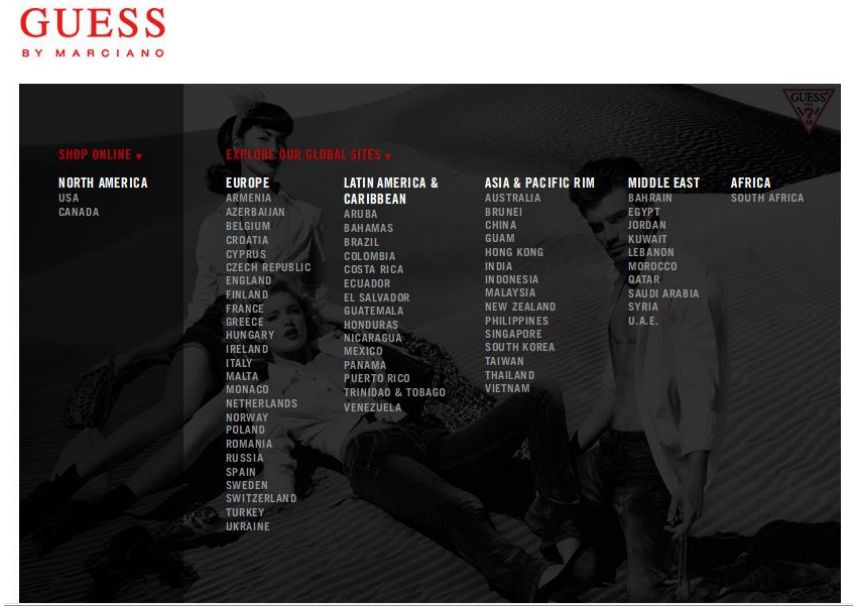
Sniffing

- njuškanje za zaporkama
- programi koji prate mrežni promet s ciljem "hvatanja" dijelova u kojima korisnik unosi svoje korisničko ime i zaporku

SQL Injection

- Umetanje SQL-meta znakova u korisnički upit s ciljem izvršavanja upita u *back-end* sustavu baze
- Realizira se kroz slanje upita s jednostrukim navodnikom (‘) – rezultat je poruka s detaljnim informacijama o *back-end* sustavu baze ili čak i omogućen pristup *back-end* dijelovima baze (uvijek ispunjiv Boolean upit)

SQL Injection – primjer napada

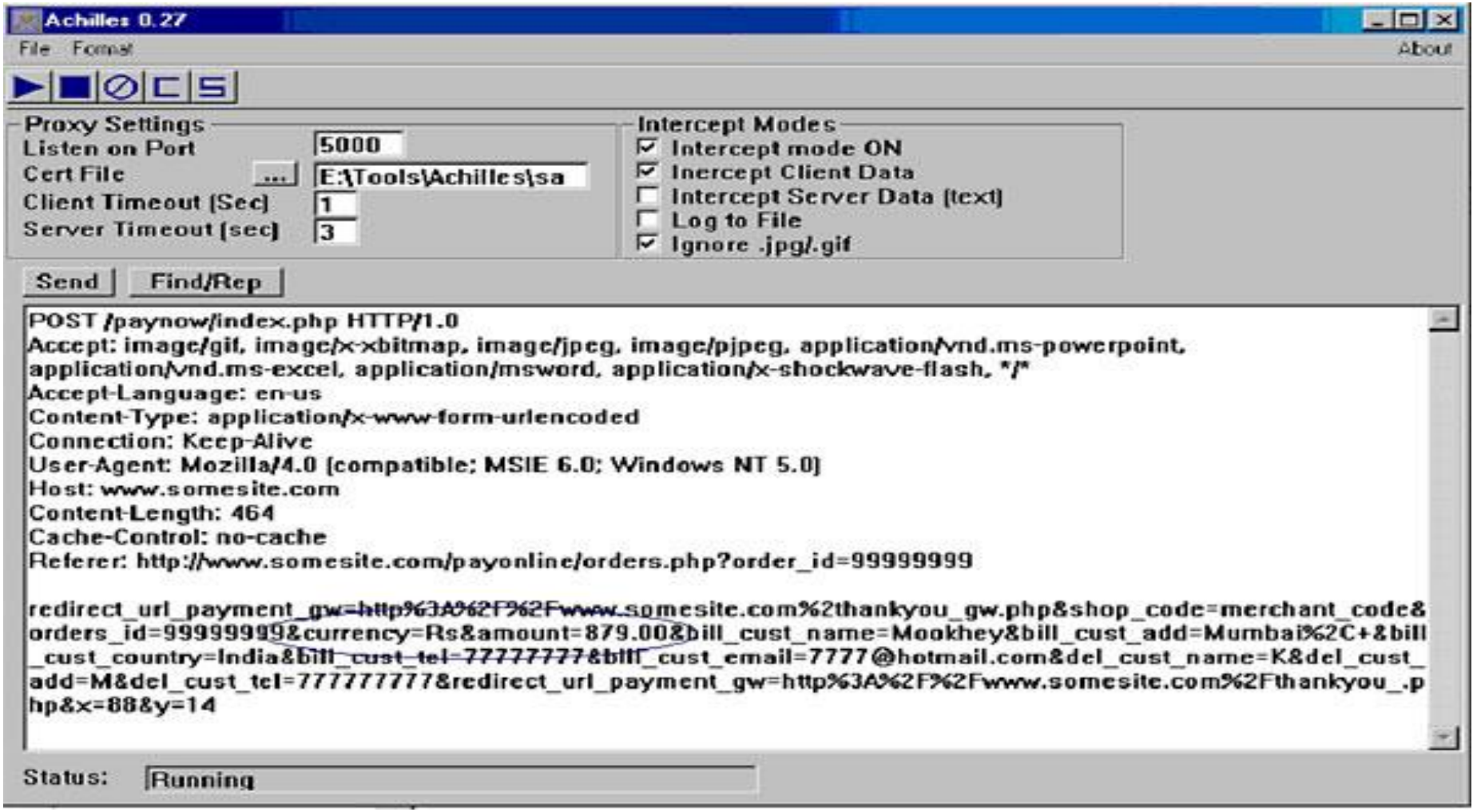


SQL Injection – primjer napada

Manipulacija cijenama

- Napad karakterističan samo za e-poslovanje
- Ukupan iznos za plaćanje se sprema u skriveno polje dinamički kreirane web stranice
- Napadač koristi proxy poslužitelj web aplikacije kako bi izmijenio iznos u trenutku njegove razmjene između web poslužitelja i korisničkog web preglednika

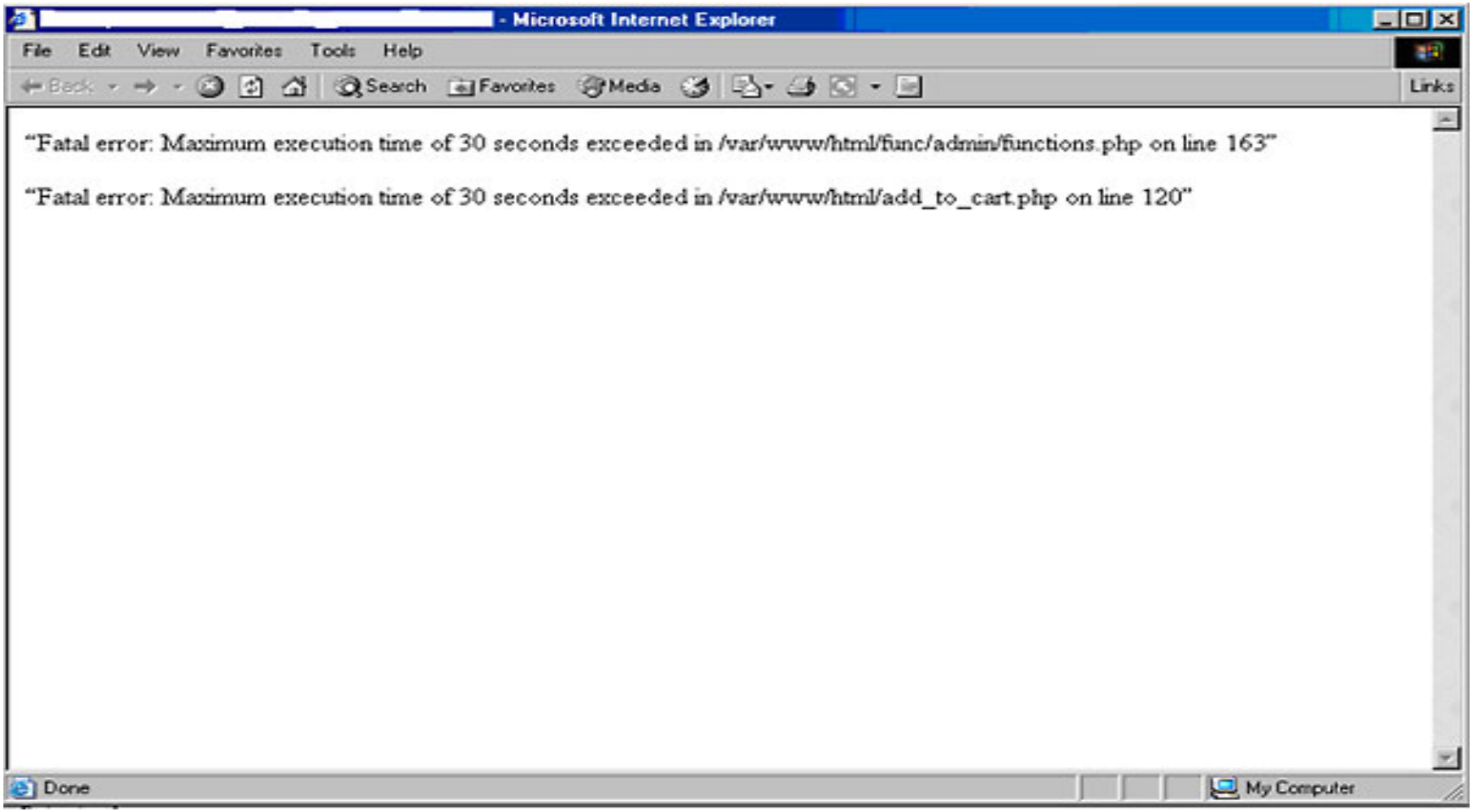
Manipulacija cijenama - primjer



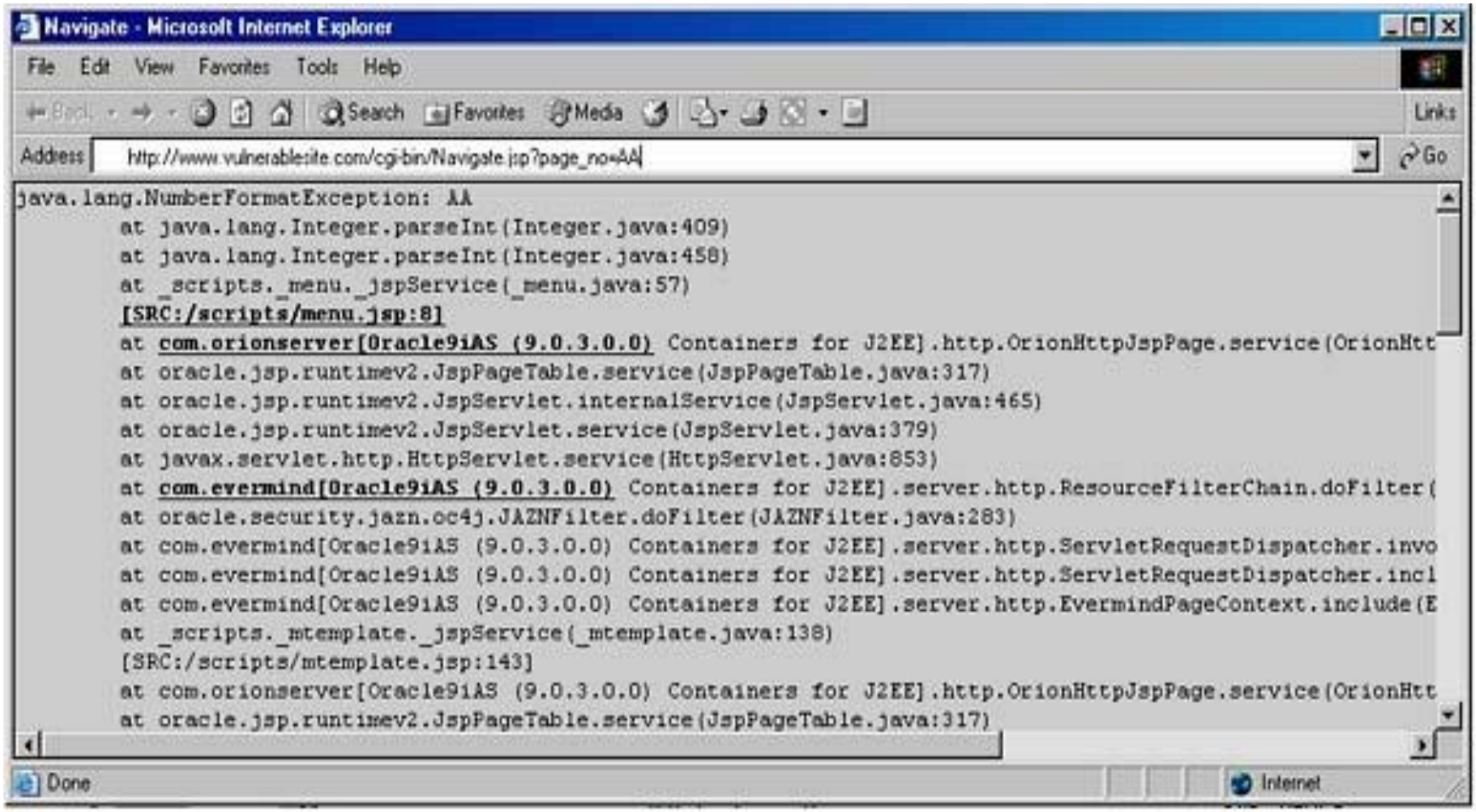
Buffer overflow

- Slanje velike količine podataka (bitova) web aplikacijama koje nisu razvijene s ciljem obrade velikih količina podataka rezultira neželjenim posljedicama.
- Cilj napada je dobiti informaciju o pogrešci koja najčešće sadrži detaljnu putanju *admin* dijela poslužitelja.

Buffer overflow – PHP time out



Buffer overflow – iskorištavanje poruke o pogrešci za daljnje napade



XSS

- *Cross-site scripting*
- Napad usmjeren na krajnjeg korisnika
- Temelji se na:
  - Nedostatnoj verifikaciji od strane web aplikacije
  - Povjerenju krajnjeg korisnika u URL web mjesta koje je napadnuto

XSS

- Korisnik unosi podatke u web formu, koja ih obrađuje i ispisuje ih na web stranici zajedno sa originalnim korisničkim unosom.
- Ako nad podacima nije izvršen *parsing* napadač može umetnuti dio JavaScript koda kao korisnički unos.
- Takvim kodom moguće je modificirati URL, a korisnik ne sluteći prijevaru jednim klikom na “provjereni URL” pokreće izvođenje skripte na svom računalu.

XSS

- Najčešći motiv XSS napada je:
  - Ukrasti *cookie* (zbog informacija o sesiji koje on sadrži)
  - Preusmjeriti korisnika na napadačevo web mjesto s kojeg će se izvršiti maliciozni kod pomoću ActiveX kontrola
  - Realizirati *phishing* prijevaru – preusmjeriti korisnika na web stranicu koja izgleda kao originalna, ali to nije.