Bezbjednost računarskih mreža

- Šta je mrežna sigurnost?
- Principi kriptografije
- Integritet poruke, autentifikacija
- Sigurnost elektronske pošte
- Sigurnost TCP konekcija: TLS
- Sigurnost na mrežnom sloju: IPsec
- Sigurnost u bežičnim i mobilnim mrežama
- Sigurnost u praksi: firewall-i i IDS

Šta je mrežna bezbjednost?

Povjerljivost: samo pošiljalac i primalac kojem je poruka namijenjena treba da „razumiju" sadržaj poruke

- Pošiljalac enkriptuje poruku
- Primalac dekriptuje poruku

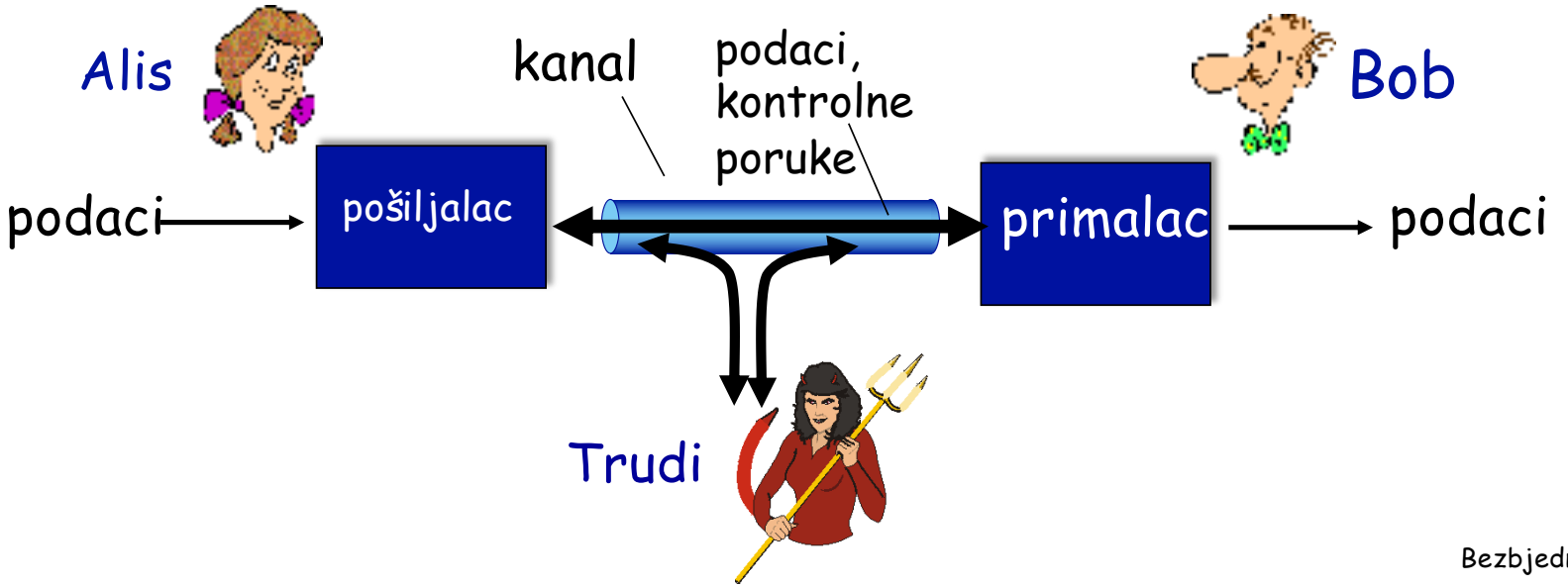
Autentifikacija: pošiljalac i primalac žele da međusobno potvrde svoje identitete

Integritet poruke: pošiljalac i primalac žele da imaju mogućnost detekcije izmjena poruke (prilikom prenosa ili naknadno, slučajnih ili zlonamjernih)

Dostupnost: servisi moraju biti dostupni korisnicima

Prijatelji i neprijatelji: Alisa, Bob, Trudi

- Dobro poznati u svijetu bezbjednosti
- Bob, Alis žele da komuniciraju "sigurno"
- Trudi (uljez) može presreti, dodavati ili brisati poruke



Prijatelji i neprijatelji: Alisa, Bob, Trudi

Ko bi mogli biti Bob i Alisa?

- … pa, Bob i Alisa iz realnog života!
- Web pretraživač/server za elektronske transakcije (npr. on-line trgovina)
- Klijent/server u on-line bankarstvu
- DNS serveri
- BGP ruteri koji razmjenjuju informacije o dostupnosti mrežnih prefiksa
- Drugi primjeri?

Postoje i „loši momci“ u ovoj priči!

Šta mogu da rade „loši momci“?

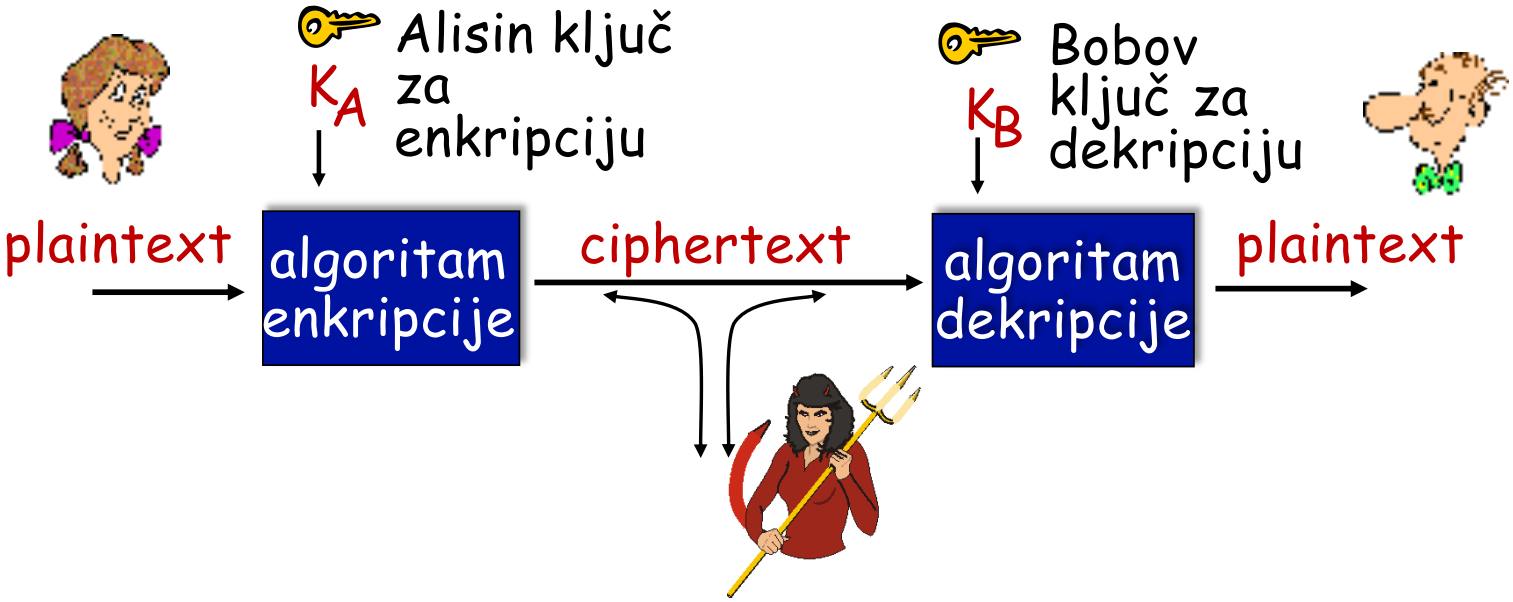
Mnogo toga!

- Prisluškivanje: presrijetanje poruka
- Aktivno ubacivanje poruka u konekciju
- Lažno predstavljanje: mogu lažirati (eng. *spoof*) izvorišnu adresu u paketu (ili bilo koje polje u paketu)
- *Hijacking*: “oteti” tekuću konekciju ubacujući se umjesto pošiljaoca ili primaoca
- Onemogućavanje servisa: spriječiti da servis bude dostupan drugima (npr. preopterećivanjem resursa)

Bezbjednost računarskih mreža

- Šta je mrežna sigurnost?
- Principi kriptografije
- Integritet poruke, autentifikacija
- Sigurnost elektronske pošte
- Sigurnost TCP konekcija: TLS
- Sigurnost na mrežnom sloju: IPsec
- Sigurnost u bežičnim i mobilnim mrežama
- Sigurnost u praksi: *firewall*-i i IDS

Jezik kriptografije



m: plaintext (originalna poruka)

$K_A(m)$: ciphertext (šifrat) - poruka enkriptovana ključem K_A

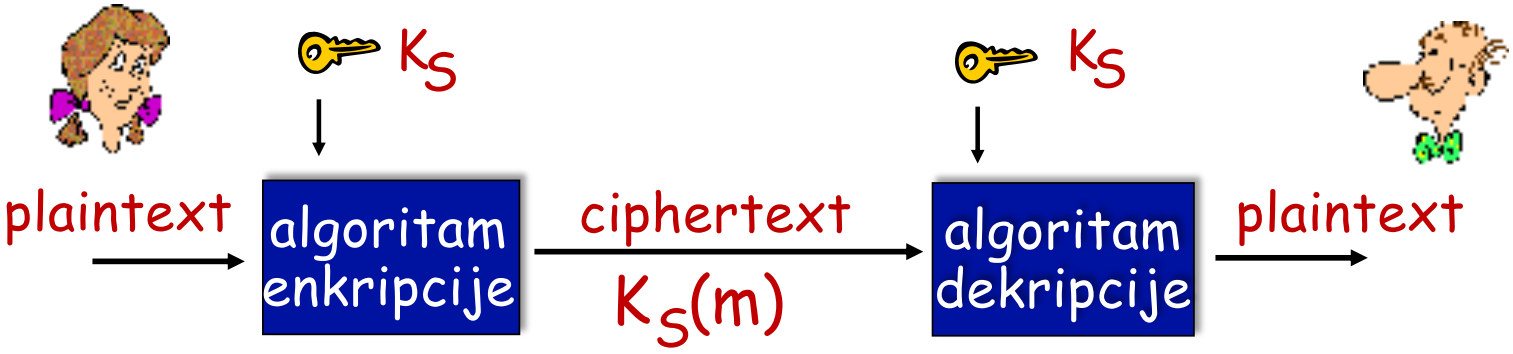
$m = K_B(K_A(m))$

„Razbijanje" enkripcije

- *Cipher-text only* napad: Trudi analizira ciphertext
- Dva pristupa:
  - *Brute force*: pretraga svih ključeva
  - Statistička analiza
- *Known-plaintext* napad: Trudi posjeduje par(ove) plaintext-ciphertext

  Npr. kod monoalfabetskog šifratora Trudi određuje mapiranja za a,l,i,c,e,b,o,
- *Chosen-plaintext* napad: Trudi može dobiti ciphertext za odabrani plaintext

Simetrična kriptografija



Simetrični ključ: Bob i Alisa dijele isti ključ: K

npr. ključ je poznati obrazac supstitucije kod monoalfabetskog šifratora

P: Kako se Bob i Alisa dogovaraju oko vrijednosti ključa?

Jednostavna šema enkripcije

Substitucioni šifrator: substitucija (zamjena) jedne stvari drugom

Monoalfabetski šifrator: zamijeniti jedno slovo drugim

plaintext: abcdefghijklmnopqrstuvwxyz

ciphertext: mnbvcxzasdfghjklpoiuytrewq

npr:

Plaintext: bob. i love you. alice

ciphertext: nkn. s gktc wky. mgsbc

Ključ enkripcije: mapiranje seta od 26 slova u drugi set od 26 slova

Sofisticiraniji metod enkripcije

- n substitucionih šifratora, $M_1, M_2, \ldots, M_n$
- ciklični obrazac:
  - npr., n=4: M_1, M_3, M_4, M_3, M_2; M_1, M_3, M_4, M_3, M_2; ..
- za svaki novi plaintext simbol koristi se sledeći supstitucioni patern po cikličnom redosledu
  - dog: d prema M_1, o prema M_3, g prema M_4

- Ključ enkripcije: n supstitucionih šifratora i ciklični obrazac
  - Ključ ne mora biti samo n-bitni obrazac

DES algoritam simetrične enkripcije

DES: Data Encryption Standard

- US standard za enkripciju [NIST 1993]
- 56-bitni simetrični ključ, 64-bitni plaintext ulaz
- Blokovski šifrator sa lančanim uvezivanjem ciphertext blokova
- Koliko je DES siguran?
  - DES izazov: Fraza enkriptovana 56-bitnim ključem dekriptovana (brute force) za manje od jednog dana
  - Nema poznatih uspješnih kriptoanalitičkih napada
- Kako poboljšati DES sigurnost:
  - 3DES: enkriptuj poruku 3 puta sa 3 različita ključa

AES: Advanced Encryption Standard

- NIST standard za simetrični enkripciju, zamijenio DES (Novembar 2001)
- Obrađuje podatke u blokovima od 128 bita
- Ključ dužine 128, 192, ili 256 bita
- *Brute force* dekripcija (probanje svih ključeva) koja zahtijeva 1s kod DES-a, zahtijeva 149 triliona godina kod AES-a

Kriptografija sa javnim ključem

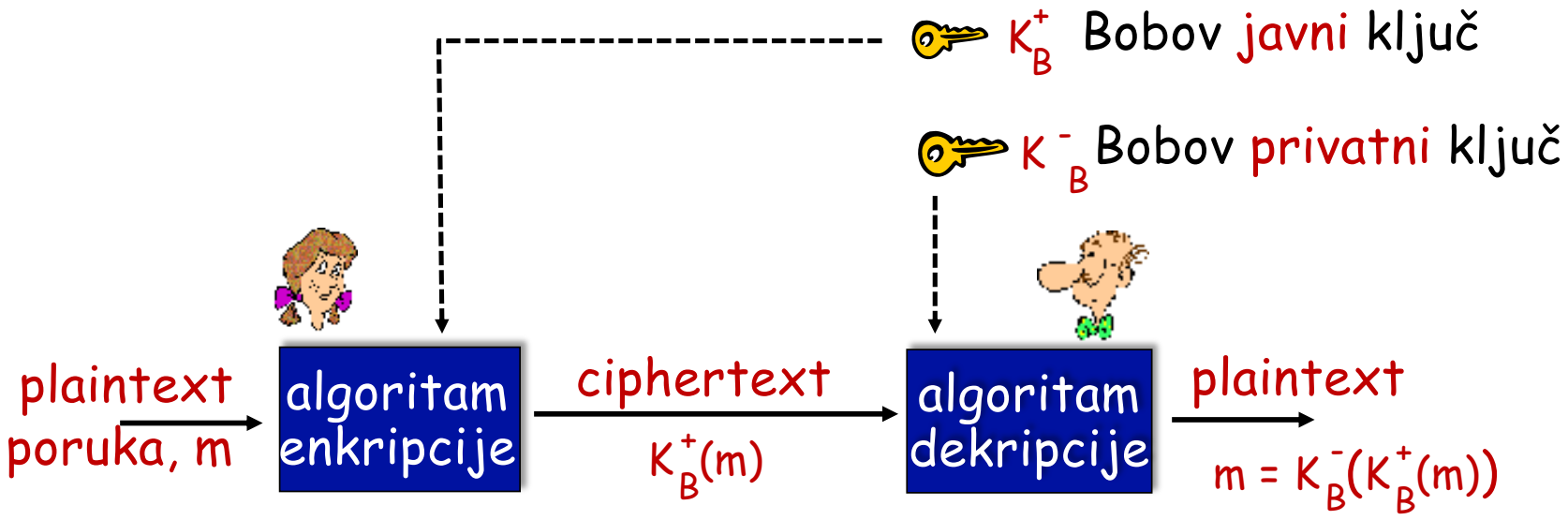
Simetrična kriptografija:

- Zahtijeva da pošiljac i primalac dijele tajni ključ
- Kako se pošiljalac i primalac dogovaraju oko vrijednosti ključa na početku (posebno ako se nikada nisu "sreli")?

Kriptografija sa javnim ključem

- Radikalno drugačiji pristup
- Pošiljalac i primalac ne dijele tajni ključ
- Javni ključ za enkripciju poznat svima
- Privatni ključ za dekripciju poznat samo primaocu

Kriptografija sa javnim ključem



- Kriptografija sa javnim ključem je totalno izmijenila 2000 godina star koncept (sa simetričnim ključem) kriptografije!
- Slične ideje su se pojavile skoro u isto vrijeme nezavisno u US i UK.

Algoritmi enkripcije javnim ključem

Zahtjevi:

1. potrebni su $K_B^+(\cdot)$ $K_B^-(\cdot)$ takvi da:

$$K_B^-(K_B^+(m)) = m$$

2. na osnovu javnog ključa K_B^+, trebalo bi da je nemoguće izračunati privatni ključ K_B^-

RSA: Rivest, Shamir, Adelson algoritam

Preduslov: aritmetika po modulu n

- x mod n = ostatak pri dijeljenju x sa n
- činjenice:
  [(a mod n) + (b mod n)] mod n = (a+b) mod n
  [(a mod n) - (b mod n)] mod n = (a-b) mod n
  [(a mod n) * (b mod n)] mod n = (a*b) mod n
- stoga:
  $(a \bmod n)^d \bmod n = a^d \bmod n$
- primjer: x=14, n=10, d=2:
  $(x \bmod n)^d \bmod n = 4^2 \bmod 10 = 6$
  $x^d = 14^2 = 196 \quad x^d \bmod 10 = 6$

RSA: Uvod

- Poruka: sekvenca bita
- Sekvenca bita se može na jedinstven način predstaviti cijelim brojem
- Stoga, enkripcija poruke ekvivalentna je enkripciji broja

Primjer:

- m= 10010001. Ova poruka se jedinstveno može predstaviti sa decimalnim brojem 145.
- u cilju enkripcije m, enkriptovaće se odgovarajući broj, što će rezultirati novim brojem (ciphertext-om).

RSA: Kreiranje para ključeva (javni i privatni)

1. odabrati dva velika prosta broja p, q. (npr., 1024 bita dužine)
2. izračunati n = pq, z = (p-1)(q-1)
3. odabrati e (pri čemu je e<n) koji nema zajedničkih faktora sa z (e, z su "uzajamno prosti").
4. odabrati d takvo da je ed-1 djeljivo sa z bez ostatka. (drugim riječima: ed mod z = 1).
5. Javni ključ je (n,e). Privatni ključ je (n,d).

K_B^+ K_B^-

RSA: enkripcija, dekripcija

0. za zadate (n,e) i (n,d) koji su prethodno izračunati

1. Poruka m (<n) se enkriptuje izračunavanjem

$$c = m^e \bmod n$$

2. Ciphertext c se dekriptuje izračunavanjem

$$m = c^d \bmod n$$

$$m = (\underbrace{m^e \bmod n}_{c})^d \bmod n$$

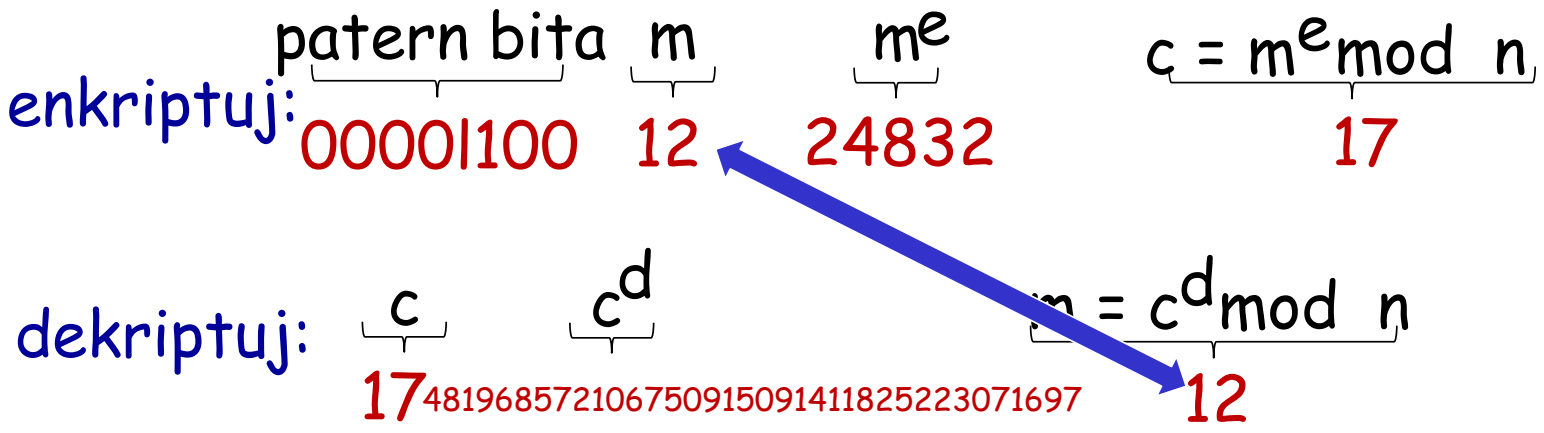
RSA primjer:

Bob bira p=5, q=7. To znači da je n=35, z=24.

e=5 (e i z su uzajamno prosti).

d=29 (ed-1 treb da je djeljivo sa z).

Enkripcija 8-bitne poruke:



Zbog čega RSA funkcioniše?

- Mora se pokazati da je $c^d \bmod n = m$, pri čemu je $c = m^e \bmod n$
- Činjenica: za bilo koje x i y: $x^y \bmod n = x^{(y \bmod z)} \bmod n$
  - gdje je $n = pq$ i $z = (p-1)(q-1)$
- Prema tome

$$
\begin{aligned}
c^d \bmod n &= (m^e \bmod n)^d \bmod n \\
&= m^{ed} \bmod n \\
&= m^{(ed \bmod z)} \bmod n \\
&= m^1 \bmod n \\
&= m
\end{aligned}
$$

RSA: još jedna važna osobina

Sledeća osobina će biti veoma važna kasnije:

$$K_B^-(K_B^+(m)) = m = K_B^+(K_B^-(m))$$

Prvo se koristi javni ključ, a zatim privani ključ

Prvo se koristi privatni ključ, a zatim javni ključ

Rezultat je isti!

Zbog čega je $K_B^-(K_B^+(m)) = m = K_B^+(K_B^-(m))$?

Slijedi direktno iz aritmetike po modulu n:

$$(m^e \bmod n)^d \bmod n = m^{ed} \bmod n$$
$$= m^{de} \bmod n$$
$$= (m^d \bmod n)^e \bmod n$$

Zbog čega se RSA smatra sigurnim?

- Pretpostavimo da je poznat Bobov javni ključ (n,e). Koliko je teško odrediti d?
- Potrebno je odrediti korišćene faktore za n bez poznavanja vrijednosti p i q
  - Faktorisanje velikih brojeva je izuzetno računski zahtjevno!

RSA u praksi: ključevi sesije

- Izračunavanje eksponenta u RSA proračunima je računski zahtjevno
- DES je barem 100 puta brži od RSA
- Preporuka je koristiti kriptografiju sa javnim ključem za uspostavljanje sigurne konekcije, a zatim razmijeniti drugi ključ - simetrični ključ sesije - za enkripciju podataka
- Ključ sesije, K_S
  - Bob i Alisa koriste RSA da razmijene simetrični ključ sesije K_S
  - Nakon što obije strane poznaju K_S, koriste simetričnu kriptografiju za šifrovanje podataka

Bezbjednost računarskih mreža

- Šta je mrežna sigurnost?
- Principi kriptografije
- Integritet poruke, autentifikacija
- Sigurnost elektronske pošte
- Sigurnost TCP konekcija: TLS
- Sigurnost na mrežnom sloju: IPsec
- Sigurnost u bežičnim i mobilnim mrežama
- Sigurnost u praksi: firewall-i i IDS

Autentifikacija

Cilj: Bob želi da mu Alisa "dokaže" svoj identitet

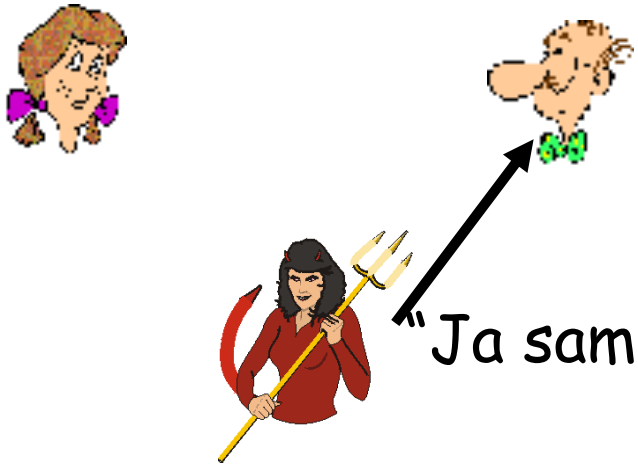
Protokol ap1.0: Alisa kaže "Ja sam Alis"

Mogući problemi??

Autentifikacija

Cilj: Bob želi da mu Alisa "dokaže" svoj identitet

Protokol ap1.0: Alisa kaže "Ja sam Alisa"

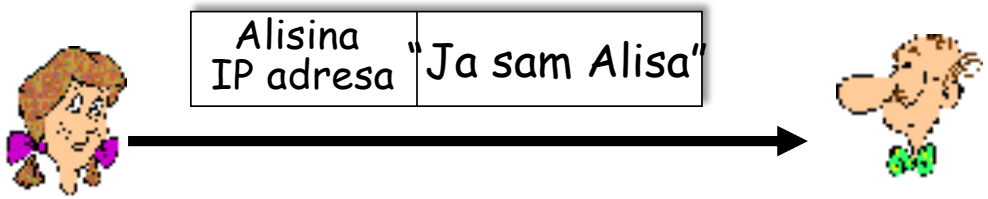


U mreži, Bob ne može „vidjeti" Alisu, pa se Trudi jednostavno može lažno predstaiti kao Alisa

Autentifikacija: drugi pokušaj

Cilj: Bob želi da mu Alisa "dokaže" svoj identitet

Protokol ap2.0: Alisa kaže "Ja sam Alisa" u IP paketu koji sadrži njenu IP adresu

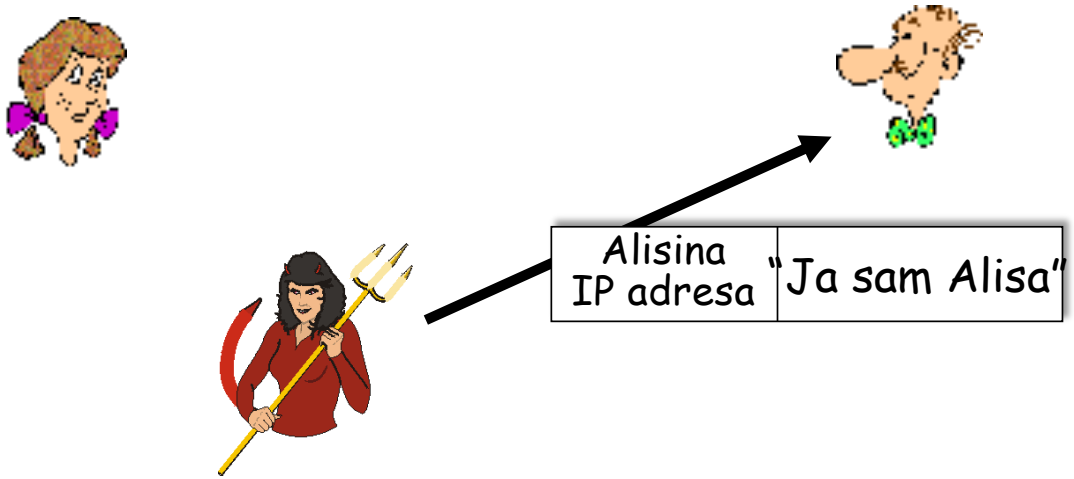


Mogući problemi??

Autentifikacija: drugi pokušaj

Cilj: Bob želi da mu Alisa "dokaže" svoj identitet

Protokol ap2.0: Alis kaže "Ja sam Alisa" u IP paketu koji sadrži njenu IP adresu

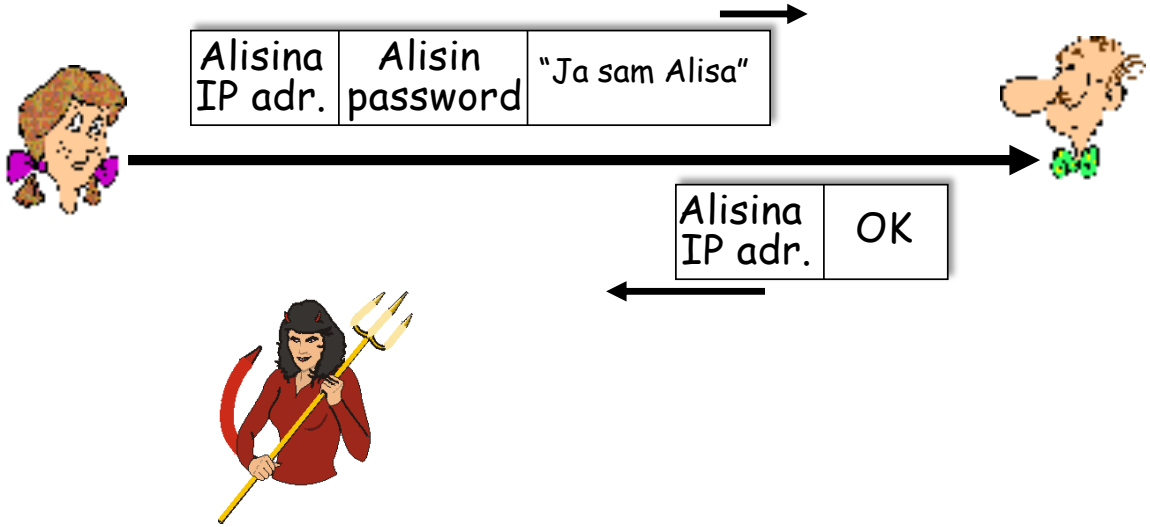


Trudi može kreirati paket sa Alisinom IP adresom („*spoofing*")

Autentifikacija: treći pokušaj

Cilj: Bob želi da mu Alisa "dokaže" svoj identitet

Protokol ap3.0: Alisa kaže "Ja sam Alisa" i šalje svoj tajni password da to „dokaže".

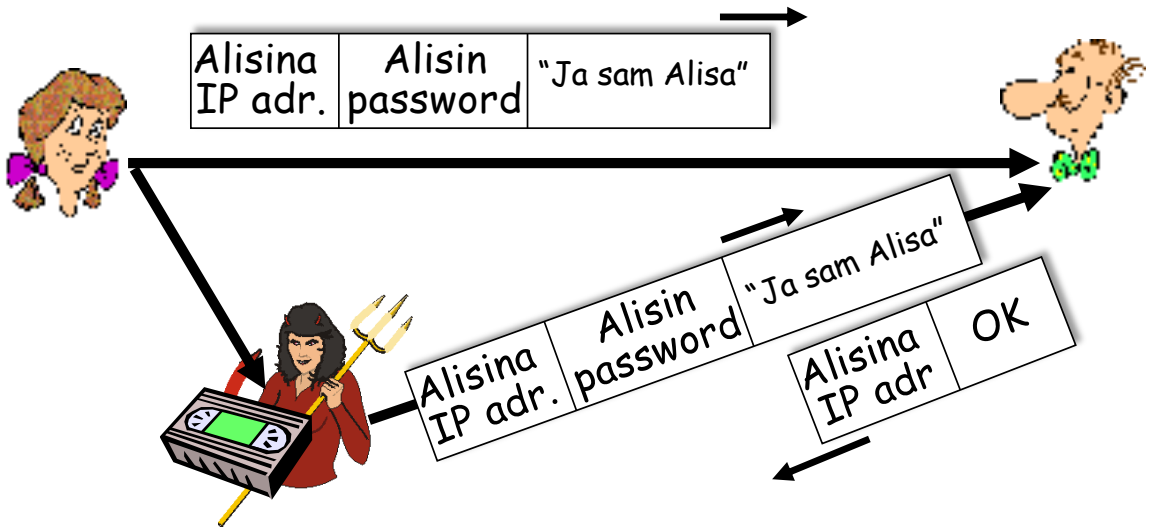


Mogući problemi??

Autentifikacija: treći pokušaj

Cilj: Bob želi da mu Alisa "dokaže" svoj identitet

Protokol ap3.0: Alis kaže "Ja sam Alisa" i šalje svoj tajni password da to „dokaže".

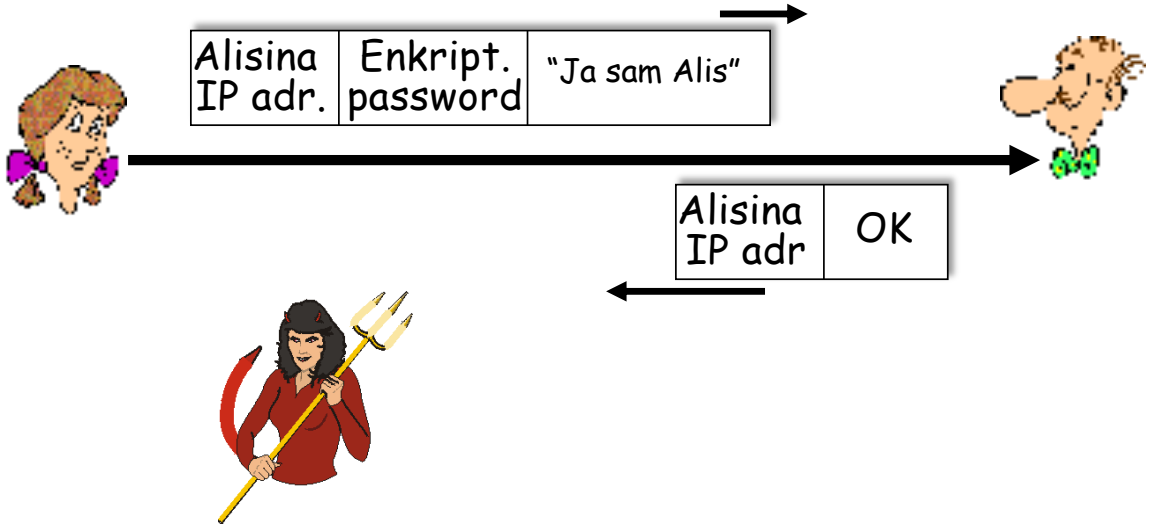


Napad ponavljanjem: Trudi snima Alisin paket tokom prenosa kanalom i šalje ga kasnije Bobu

Autentifikacija: modifikovani treći pokušaj

Cilj: Bob želi da mu Alisa "dokaže" svoj identitet

Protokol ap3.0: Alisa kaže "Ja sam Alisa" i šalje svoj enkriptovani tajni password da to „dokaže".

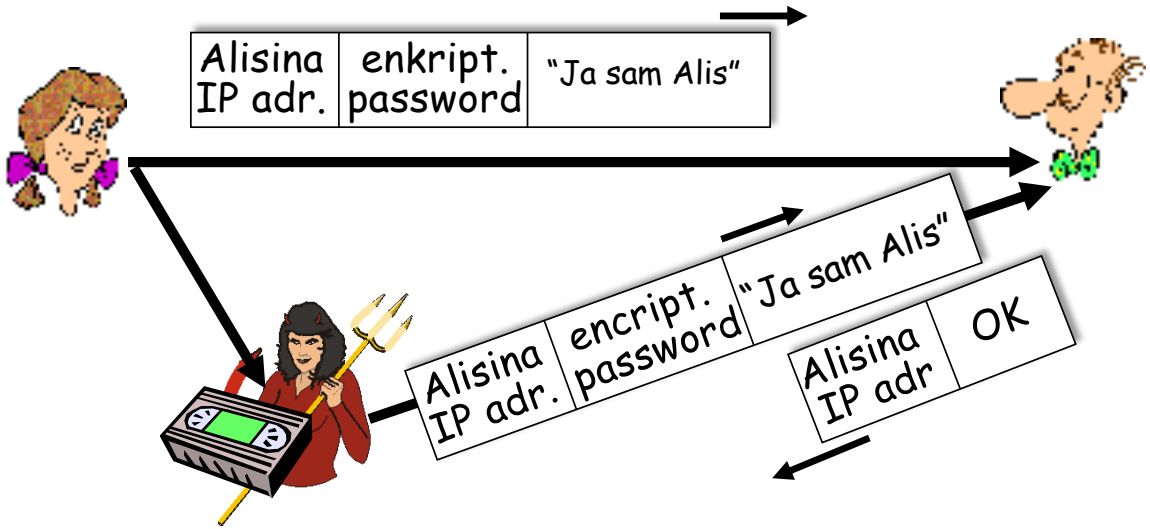


Mogući problemi??

Autentifikacija: modifikovani treći pokušaj

Cilj: Bob želi da mu Alisa "dokaže" svoj identitet

Protokol ap3.0: Alisa kaže "Ja sam Alisa" i šalje svoj enkriptovani tajni password da to „dokaže".



Napad ponavljanjem i dalje je moguć: Trudi snima Alisin paket i šalje ga naknadno Bobu

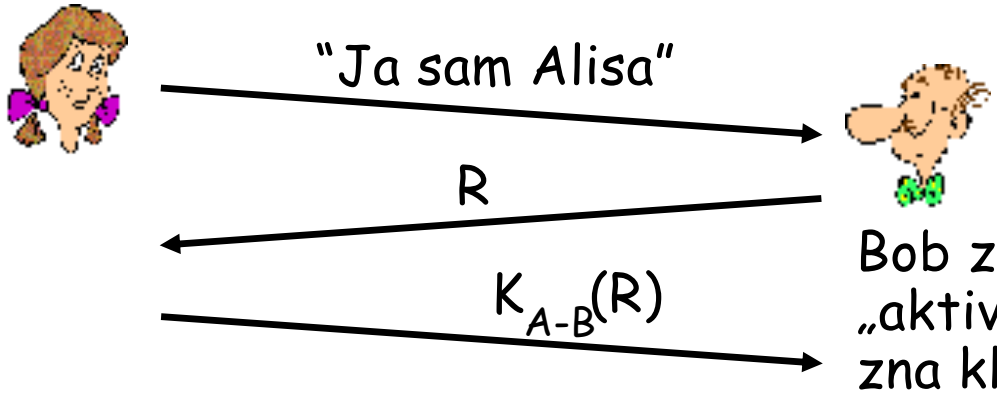
Autentifikacija: četvrti pokušaj

Cilj: izbjeći napad ponavljanjem poruke

nonce: broj (R) koji se koristi samo jednom u toku sesije

protokol ap4.0: Bob generiše i šalje nonce (R) Alisi,

- Alisa mora vratiti R enkriptovano sa dijeljenim tajnim ključem



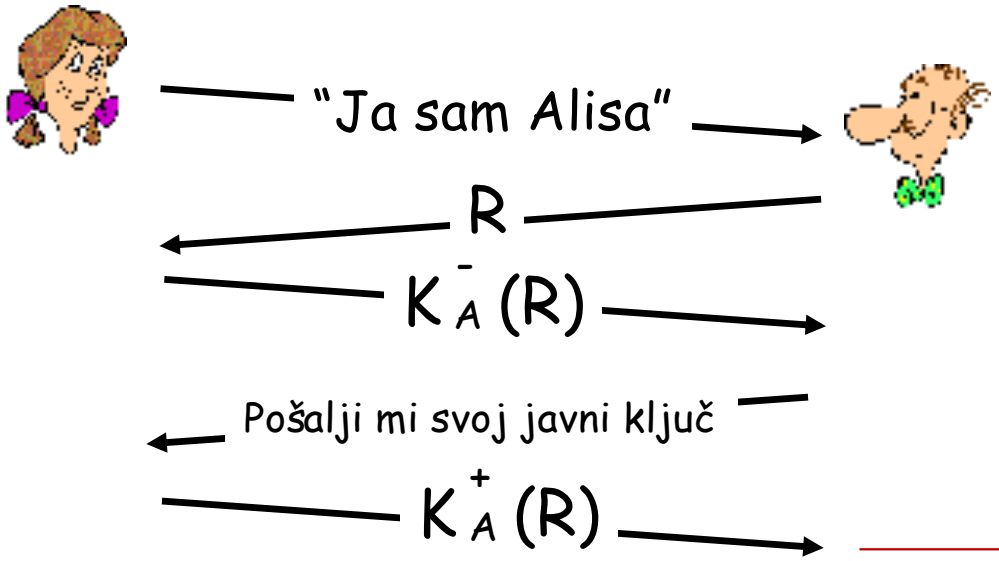
Bob zna da je Alisa „aktivna", i samo Alisa zna ključ za enkripciju nonce-a, stoga primljena poruka mora biti od Alis!

Mogući problemi, nedostaci?

Autentifikacija: ap5.0

ap4.0 zahtijeva dijeljeni simetrični ključ – da li se možemo autentifikovati pomoću tehnika javnog ključa?

ap5.0: koristiti nonce i kriptografiju javnog ključa



Bob računa

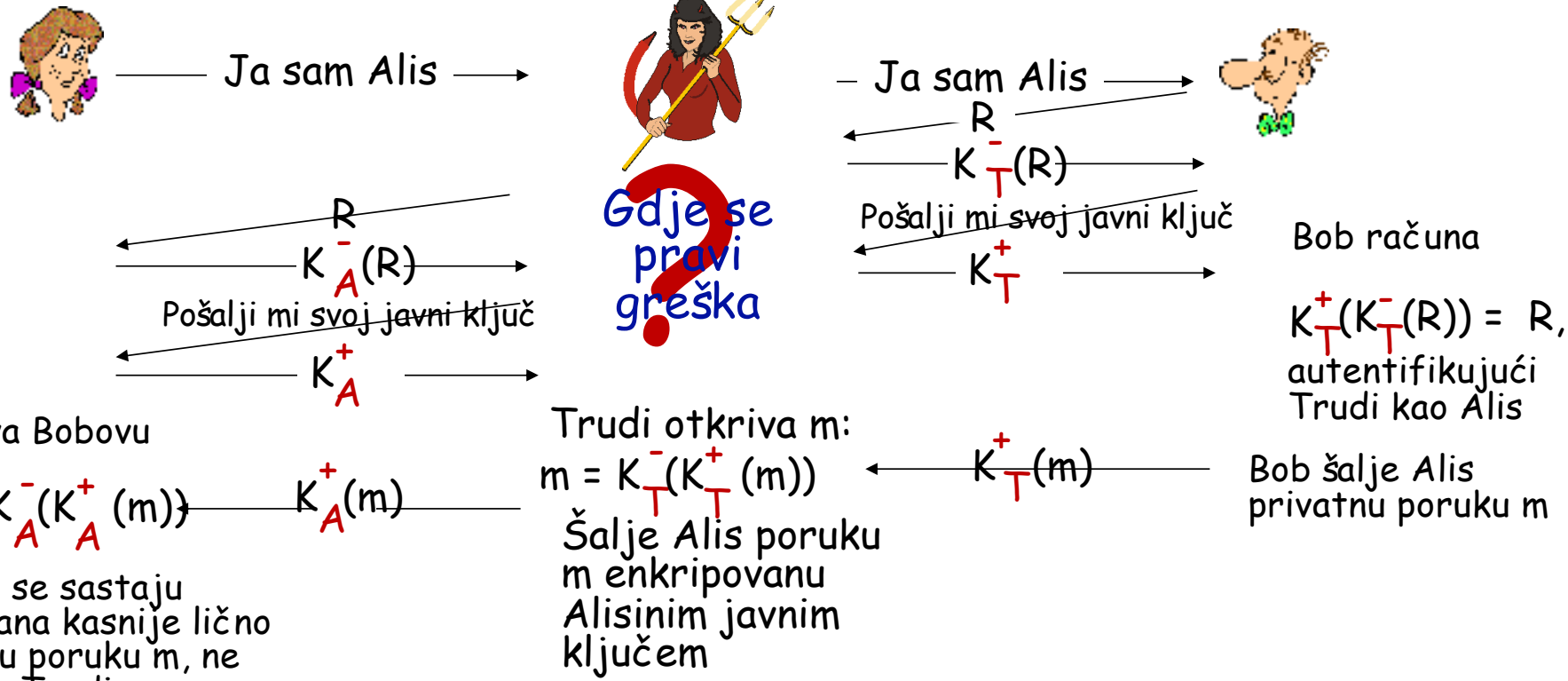
$K_A^+(K_A^-(R)) = R$

i zna da samo Alisa može imati privatni ključ koji je enkriptovao R tako da je:

$K_A^+(K_A^-(R)) = R$

Autentifikacija: ap5.0 – još uvijek postoje mane!

Man in the middle napad: Trudi se predstavlja kao Alis prema Bobu, a kao Bob prema Alis.



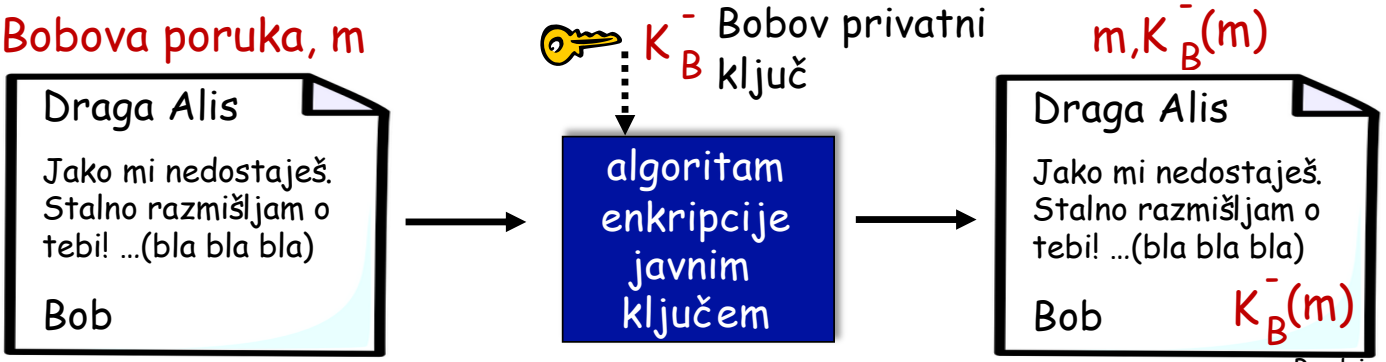
Bezbjednost računarskih mreža

- Šta je mrežna sigurnost?
- Principi kriptografije
- **Integritet poruke**, autentifikacija
- Sigurnost elektronske pošte
- Sigurnost TCP konekcija: TLS
- Sigurnost na mrežnom sloju: IPsec
- Sigurnost u bežičnim i mobilnim mrežama
- Sigurnost u praksi: firewall-i i IDS

Digitalni potpisi

Kriptografska tehnika analogna rukom pisanim potpisima:

- Pošiljalac (Bob) digitalno potpisuje dokument: on je vlasnik/kreator dokumenta.
- Provjerljivo, ne može se krivotvoriti: primalac (Alisa) može dokazati nekom da je Bob, a ne neko drugi, potpisao dokument
- Jednostavan digitalni potpis za poruku m:
  - Bob potpisuje m enkriptujući svojim privanim ključem K_B, kreirajući "potpisanu" poruku, $K_B^-(m)$



Digitalni potpisi

- Pretpostaviti da Alisa prima potpisanu poruku m – $K_B^-(m)$
- Alis provjerava da li je poruka potpisana od strane Boba tako što primjenjuje Bobov javni ključ K_B^+ nad $K_B^-(m)$ i provjerava da li je $K_B^+(K_B^-(m)) = m$.
- Ako je $K_B(K_B(m)) = m$, onaj ko je potpisao m morao je imati Bobov privatni ključ

Alisa stoga može provjeriti:

- da li je Bob potpisao m
- da niko drugi nije potpisao m
- da je Bob potpisao m a nije m'

Nemogućnost poricanja:

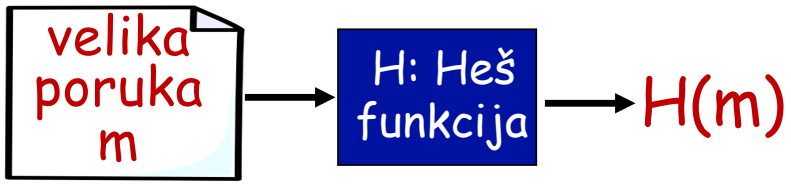
- ✓ Alisa može sačuvati potpisanu poruku $K_B^-(m)$ i dokazati na sudu da je Bob potpisao m

Sažetak poruke (*Message digest*)

Računski je kompleksno potpisati cijelu poruku

cilj: digitalni „otisak" (fingerprint) fiksne dužine, lak za računanje

- Primjenjuje se heš funkcija H na m, dobija se heš vrijednost (*message digest*) H(m)



Osobine heš funkcije:

- many-to-1 mapiranje
- proizvodi heš vrijednosti (sažetak) fiksne dužine
- na osnovu heš vrijednosti x, računski je nemoguće (nepraktično) pronaći m takvo da je x = H(m)

Internet checksum: slaba kriptografska heš funkcija

Internet checksum-a ima neke osobine heš funkcije:

- proizvodi sažetak fiksne dužine (16 bita)
- *many-to-one* mapiranje

Međutim, na osnovu poruke sa izračunatim hešom, lako je naći drugu poruku koja ima isti heš:

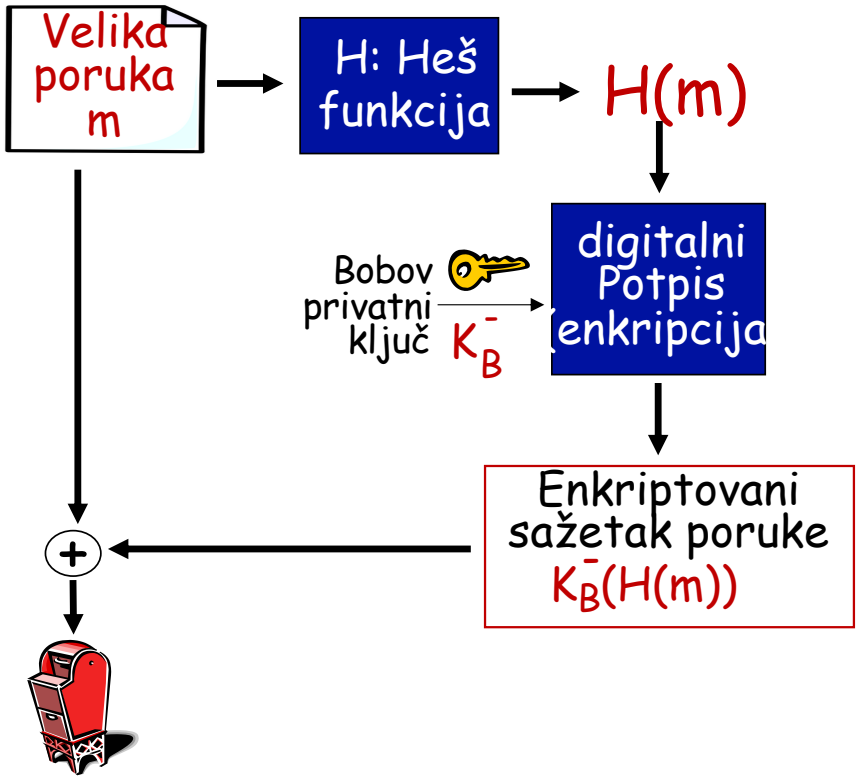
Poruka	ASCII format
I O U 1	49 4F 55 31
0 0 . 9	30 30 2E 39
9 B O B	39 42 D2 42
	B2 C1 D2 AC

Poruka	ASCII format
I O U 9	49 4F 55 39
0 0 . 1	30 30 2E 31
9 B O B	39 42 D2 42
	B2 C1 D2 AC

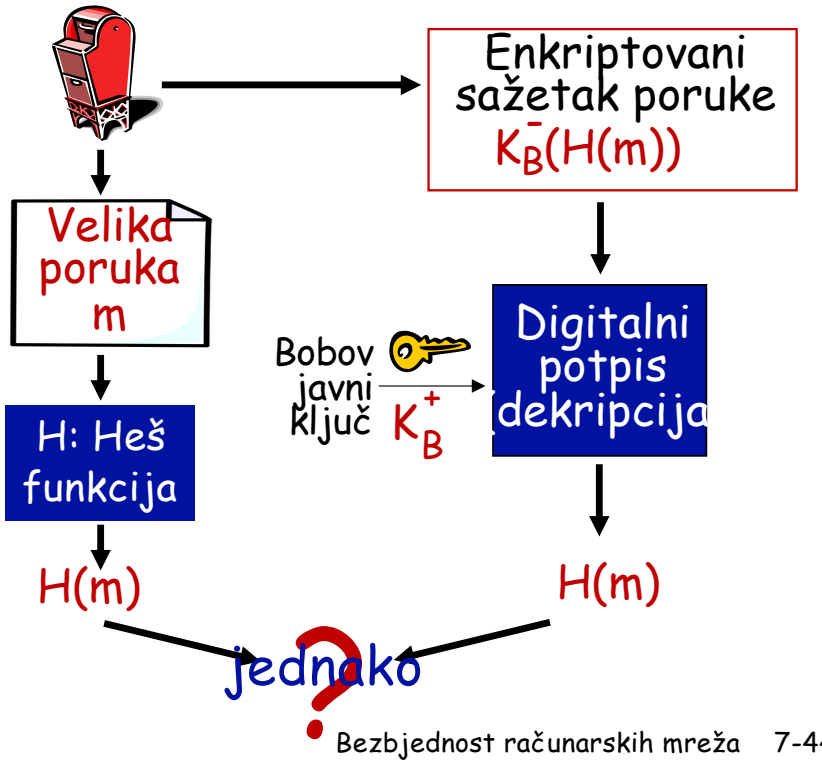
Različite poruke ali identične checksum-e!

Digitalni potpis = potpisani sažetak poruke

Bob šalje digitalno potpisanu poruku:



Alisa verifikuje potpis, integritet digitalno potpisane poruke:



Heš algoritmi

- MD5 heš funkcija (RFC 1321)
  - Računa 128-bitni sažetak poruke u 4 koraka.
  - Ranije široko zastupljen, danas se lako „razbija"
  - Napadi „heš kolizijom" ukazali na njene nedostatke pa se više ne preporučuje za upotrebu
- SHA (*Secure Hash Algorithm*)
  - NIST standardi
  - SHA-1 ima heš vrijednost od 160 bita
  - 2002. NIST je napravio revizije SHA 256, SHA 384 i SHA 512 sa hešom označenog broja bita

Autentifikacija: ap5.0 – poboljšanje!!

Podsjećanje na problem: Trudi se predstavlja Bobu kao Alisa, a Alisi kao Bob

Ja sam Alis

Gdje se pravi greška?

Ja sam Alis

R

$K_T^-(R)$

Pošalji mi svoj javni ključ

K_T^+

Bob računa $K_T^+(K_T^-(R)) = R$, autentifikujući Trudi kao Alis

R

$K_A^-(R)$

Pošalji mi svoj javni ključ

K_A^+

Alis otkriva Bobovu poruku m:

$m = K_A^-(K_A^+(m))$

$K_A^+(m)$

Ona i Bob se sastaju nedelju dana kasnije lično i diskutuju poruku m, ne znajući da Trudi zna m

Trudi otkriva m:

$m = K_T^-(K_T^+(m))$

Šalje Alisi poruku m enkripovanu Alisinim javnim ključem

$K_T^+(m)$

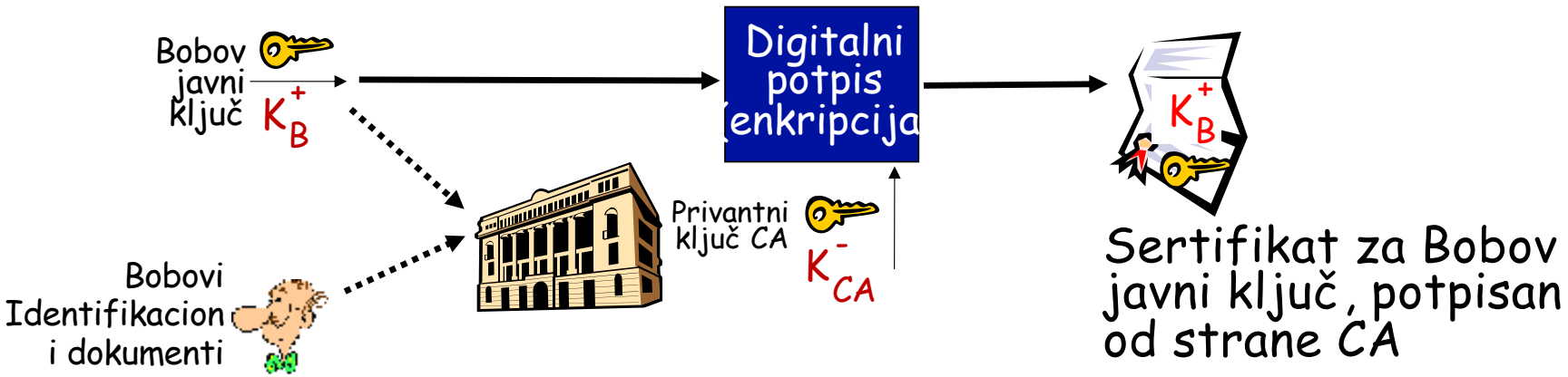
Bob šalje Alis privatnu poruku m

Potreba za sertifikovanim javnim ključevima

- Motivacija: Trudi „prenkuje" Boba prilikom porudžbine pice
  - Trudi kreira email narudžbinu
    Molim vas da mi isporučite četiri pice sa feferonima. Hvala, Bob
  - Trudi potpisuje poruku svojim privatnim ključem
  - Trudi šalje naružbinu piceriji
  - Trudi šalje piceriji svoj javni ključ, ali kaže da je to Bobov javni ključ
  - Picerija verifikuje potpis; zatim isporučuje četiri pice sa feferonima Bobu
  - Bob čak ne voli feferone

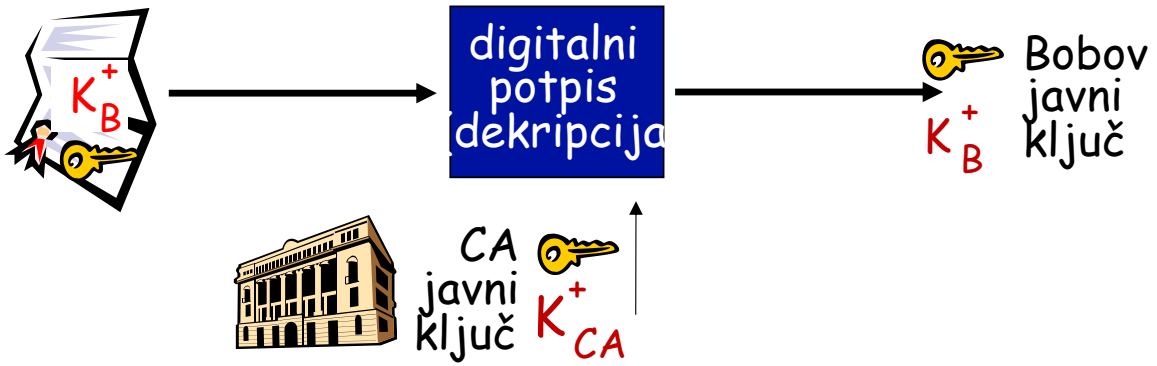
Sertifikacioni Autoriteti Javnih Ključeva

- *CA* (*Certification Authority*): vezuje javni ključ za određeni entitet E
- Entitet (osoba, web sajt, ruter) registruje svoj javni ključ dostavljajući dokaz identiteta sertifikacionom autoritetu
  - CA kreira sertifikat koji vezuje identitet entiteta E sa javnim ključem entiteta E
  - Sertifikat sadrži javni ključ entiteta E koji je digitalno potpisan od strane CA: CA na ovaj način govori "ovo je javni ključ entiteta "



Sertifikacioni Autoriteti Javnih Ključeva (CA)

- Kada Alisa želi Bobov javni ključ:
  - dobija Bobov sertifikat
  - primjenjuje javni ključ CA da bi povrdila Bobov javni ključ

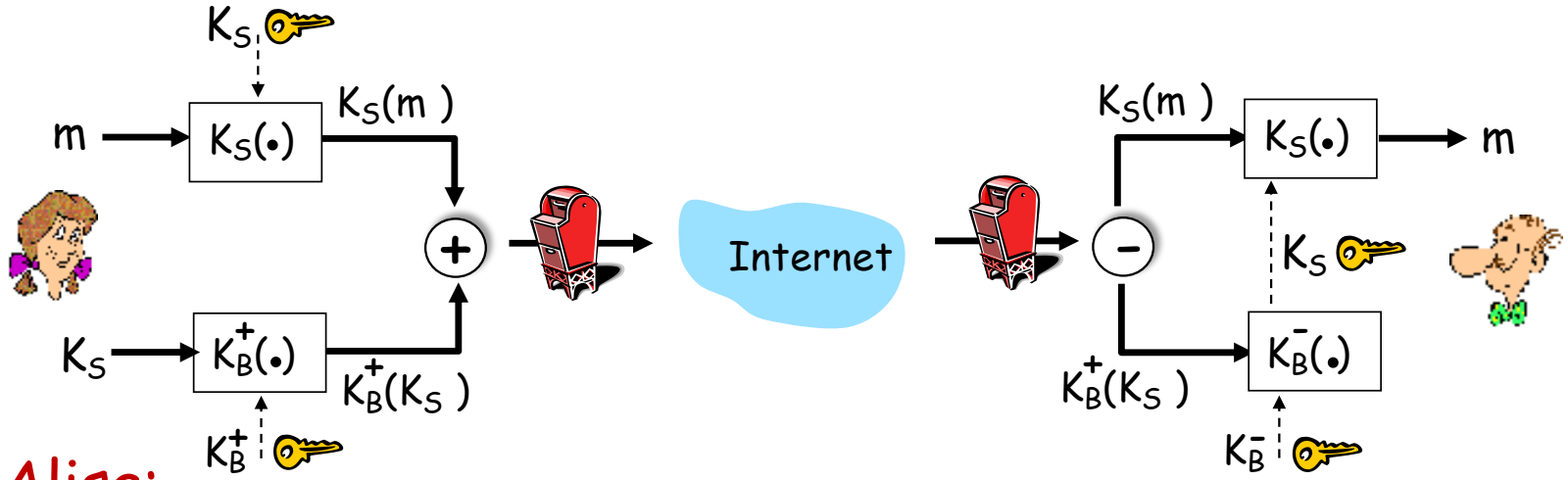


Bezbjednost računarskih mreža

- Šta je mrežna sigurnost?
- Principi kriptografije
- Integritet poruke, autentifikacija
- Sigurnost elektronske pošte
- Sigurnost TCP konekcija: TLS
- Sigurnost na mrežnom sloju: IPsec
- Sigurnost u bežičnim i mobilnim mrežama
- Sigurnost u praksi: firewall-i i IDS

Sigurni e-mail: povjerljivost

Alisa želi da pošalje povjerljiv e-mail, m, Bobu.

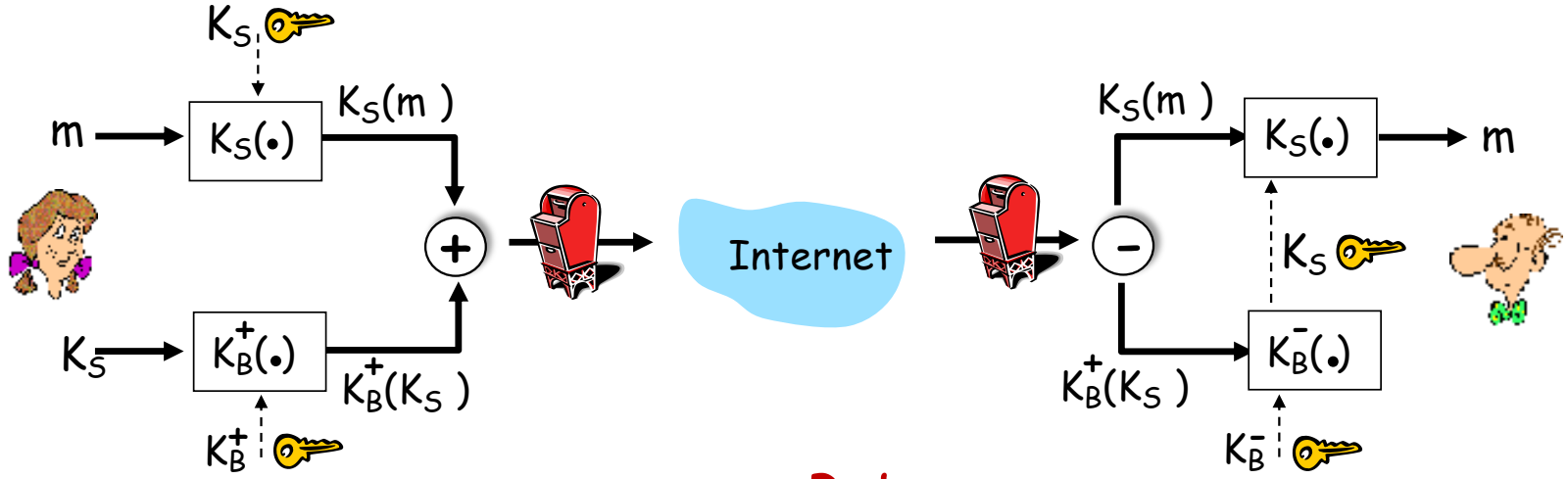


Alisa:

- generiše random simetrični privatni ključ K_S
- enkriptuje poruku sa K_S
- enkriptuje K_S sa Bobovim javnim ključem
- šalje $K_S(m)$ i $K^+_B(K_S)$ Bobu

Sigurni email: povjerljivost (više)

Alisa želi da pošalje povjerljiv e-mail, m, Bobu.

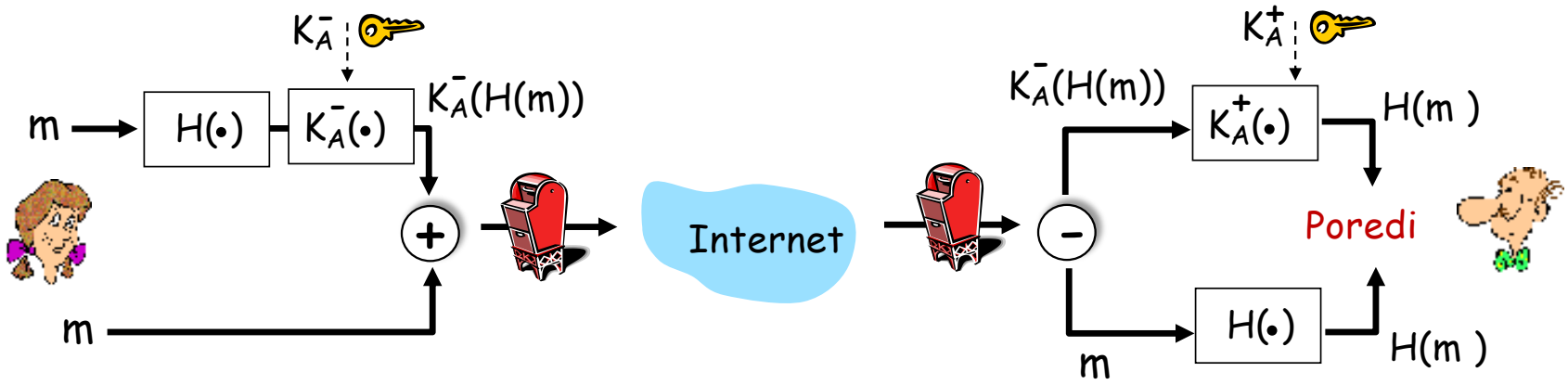


Bob:

- koristi svoj privatni ključ da dekriptuje i otkrije K_S
- koristi K_S da dekriptuje $K_S(m)$ i otkrije m

Sigurni e-mail: integritet, autentifikacija

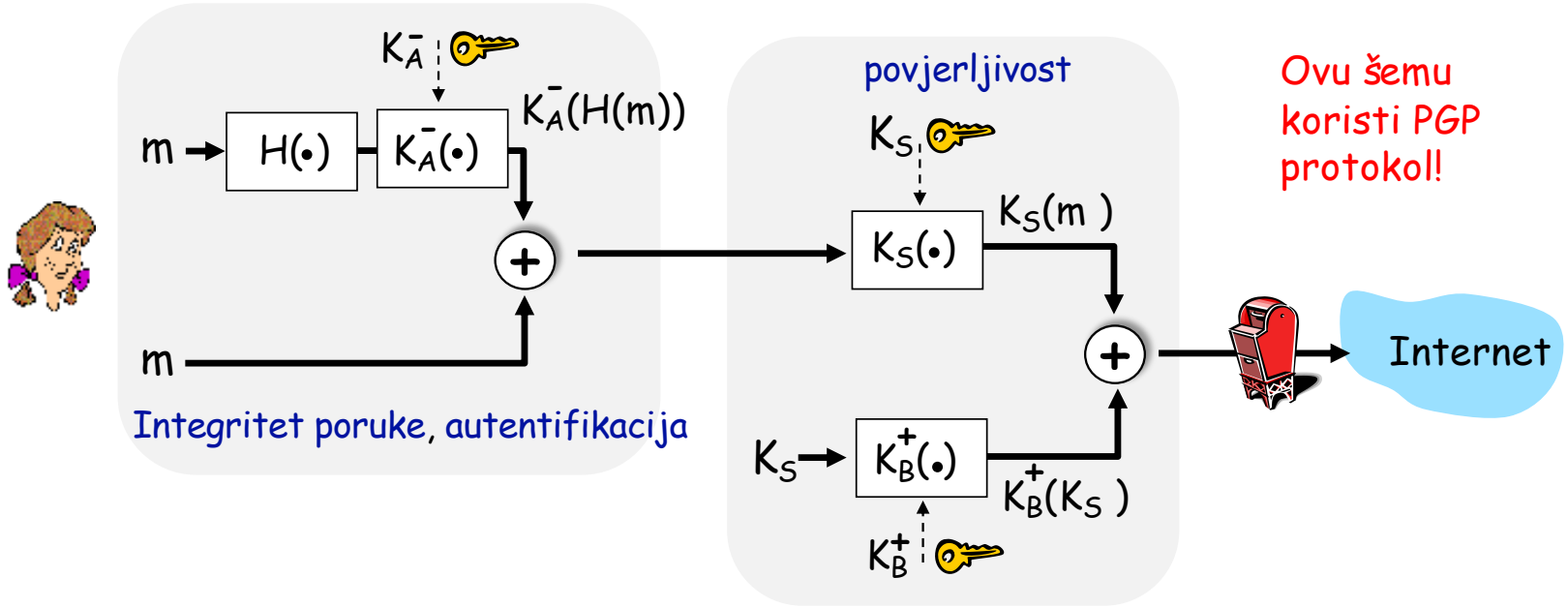
Alisa želi da pošalje m Bobu, uz potvrdu integriteta poruke i autentifikaciju



- Alis adigitalni potpisuje heš svoje poruke svojim privatnim ključem, omogućavajući provjeru integriteta i pružajući autentifikaciju
- Šalje poruku i digitalni potpis

Sigurni e-mail: integritet, autentifikacija

Alisa želi da pošalje m Bobu, uz potvrdu integriteta poruke i autentifikaciju



Alis koristi tri ključa: svoj privatni ključ, Bobov javni ključ, novi simetrični ključ

Koje akcije preduzima Bob?

Bezbjednost računarskih mreža

- Šta je mrežna sigurnost?
- Principi kriptografije
- Integritet poruke, autentifikacija
- Sigurnost elektronske pošte
- Sigurnost TCP konekcija: TLS
- Sigurnost na mrežnom sloju: IPsec
- Sigurnost u bežičnim mrežama
- Sigurnost u praksi: firewall-i i IDS

Transport-layer security (TLS)

- Široko zastupljen sigurnosni protokol iznad transportnog nivoa
  - Podržan od strane skoro svih web pretraživača, web servera: https (port 443)
- pruža:
  - tajnost: preko simetrične enkripcije
  - zaštitu integriteta: kriptografskim heširanjem
  - autentifikaciju: preko kriptografije javnog ključa

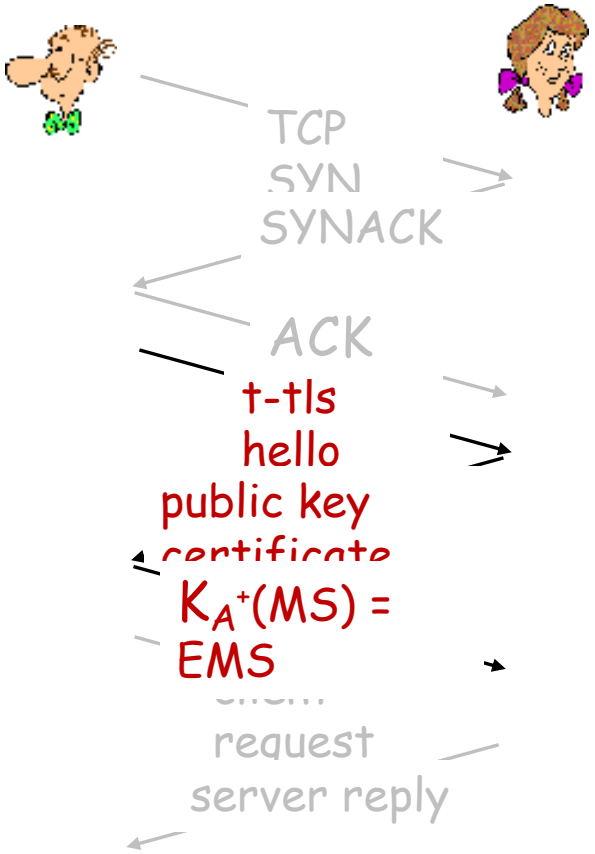
  } Sve ove tehnike smo učili!
- istorija:
  - Rana istraživanja, implementacija: sigurno mrežno programiranje, sigurni soketi
  - secure socket layer (SSL) je prevaziđen [2015]
  - TLS 1.3: RFC 8846 [2018]

TLS sigurnost: šta je potrebno?

- Razmotrimo uprošćenu implementaciju TLS protkola, t-tls, da bi se lakše moglo shvatiti koje zahtjeve TLS treba da zadovolji!

- Određeni djelovi komunikacije su već poznati:
  - Usaglašavanje (rukovanje): Alisa i Bob koriste svoje sertifikate i privatne ključeve da se međusobno autentifikuju i razmjenjuju tajnu informaciju
  - Izvođenje ključa: Alisa i Bob koriste dijeljenju tajnu informaciju da generišu set ključeva
  - Transfer podataka: razmjenjuje se tok podataka
  - Zatvaranje konekcije: specijalne poruke se koriste za sigurno zatvaranje konekcije

t-tls: Inicijalno rukovanje



t-tls faza rukovanja:

- Bob uspostavlja TCP konekciju sa Alisom
- Bob verifikuje da je Alisa stvarno Alisa
- Bob šalje Alisi master tajni ključ (MS-*Master Secret*), koji se koristi za generisanje drugih ključeva za TLS sesiju
- Potencijalni problemi:
  - 3 RTT proteknu prije nego što klijent može početi sa prijemom podataka (uključujući TCP rukovanje)

t-tls: kriptografski ključevi

- Smatra se lošom praksom korišćenje istog ključa za više od jedne kriptografske funkcije
  - Različiti ključevi za *Message Authentication Code* (MAC) i enkripciju
- Četiri ključa:
  - K_c : ključ za enkripciju podataka koji klijent šalje serveru
  - M_c : MAC ključ za podatke koje klijent šalje serveru
  - K_s : ključ za enkripciju podataka koje server šalje klijentu
  - M_s : MAC ključ za podatke koje server šalje klijentu
- Za generisanje ključeva koristi se *key derivation function* (KDF)
  - Novi ključevi se generišu na osnovu istog master ključa i (opciono) određenih random podataka

t-tls: enkripcija podataka

- TCP pruža apstrakciju toka bajta
- Mogu li se enkriptovati podaci u toku upisivanja u TCP soket?
  - Gdje se dodaje MAC kod? Ukoliko se dodaje na kraju onda se provjera intgriteta može vršiti tek nakon što se svi podaci prime i konekcija zatvori!
  - rešenje: podijeliti tok u seriju "zapisa"
    - Svaki klijent-server zapis sadrži MAC kod, kreiran na osnovu M_c
    - Prijemnik može reagovati na svaki zapis po njegovom prijemu
- t-tls zapis enkriptovan simetričnim ključem , K_c ,proslijeđen TCP-u:

t-tls: enkripcija podataka

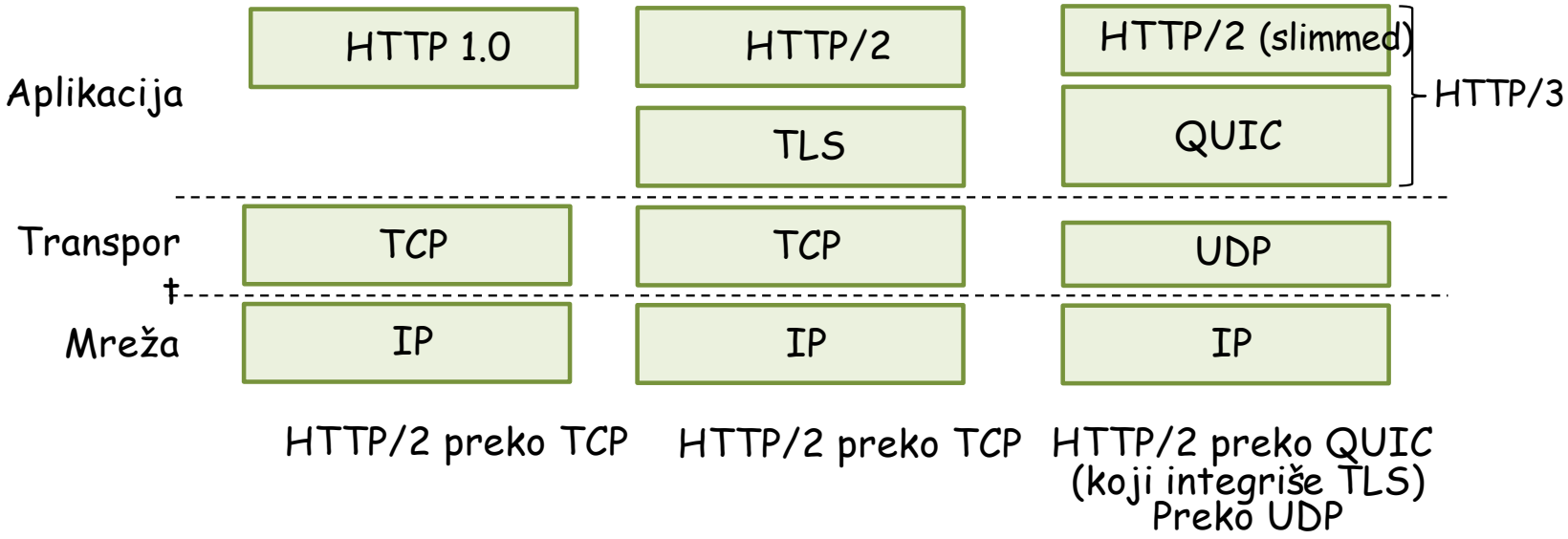
- Da li su mogući napadi na tok podataka?
  - Promjena redosleda: napadač presrijeta TCP segmente i mijenja njihov redosled (mijenjajući pritom broj u sekvenci u neenkriptovanom zaglavlju)
  - Napad ponavljanjem (*replay attack*)
- Rešenja:
  - Koristiti TLS brojeve u sekvenci (podaci, TLS-seq-# se koriste kao ulazni argumenti MAC funkcije)
  - Koristiti nonce (zaštita od ponavljanja čitave konekcije)

t-tls: zatvaranje konekcije

- Napad odsijecanjem (*truncation attack*):
  - Napadač falsifikuje segment za zatvaranje TCP konekcije
  - Jedna ili obije strane misle da je poslato manje podataka nego što je to stvarno slučaj
- Rješenje: zabilježiti informaciju o tipu zapisa, jedan tip se odnosi na zatvaranje konekcije
  - tip 0 za podatke; tip 1 za zatvaranje konekcije
- MAC se sada računa na osnovu podataka, tipa i broja u sekvenci

Transport-layer security (TLS)

- TLS pruža API koji može koristiti bilo koja aplikacija
- TLS iz ugla HTTP-a:



TLS: 1.3 Paket šifara (*cipher suite*)

- "Paket šifara": algoritmi koji se mogu koristiti za generisanje ključa, enkripciju, MAC, digitalne potpise
  - Klijent nudi izbor, server bira jedan
- TLS: 1.3 (2018): manji izbor algoritama u paketu šifara nego kod TLS 1.2 (2008)
  - samo 5 opcija, umjesto 37 opcija
  - zahtijeva se Diffie-Hellman (DH) za razmjenu ključa, umjesto da se dopušta DH ili RSA
  - Kombinovani algoritam za enkripciju i autentifikaciju ("autentifikovana enkripcija") podataka umjesto serijske enkripije i autentifikacije
    - 4 rešenja bazirana na AES-u
  - HMAC koji koristi SHA (256 ili 284) kriptografsku heš funkciju

Stvarni TLS - rukovanje

Svrha

1. Autentifikacija servera
2. Pregovaranje u vezi kripografskih algoritama
3. Dijeljenje ključeva
4. Autentifikacija klijenta (opciono)

Stvarni TLS - rukovanje

1. Klijent šalje listu algoritama koji podržava, zajedno sa odabranim jednokratnim nonce-om
2. Server bira algoritme iz liste i šalje nazad: izbor + sertifikat + svoj nonce
3. Klijent verifikuje sertifikat, izvlači javni ključ servera, generiše pre_master_secret, enkriptuje ga serverovim javnim ključem i šalje serveru
4. Klijent i server nezavisno računaju ključeve za enkripciju i MAC ključeve na osnovu pre_master_secret i nonce vrijednosti
5. Klijent šalje MAC svih poruka poslatih tokom rukovanja
6. Server šalje MAC svih primljenih poruka tokom faze rukovanja

Stvarni TLS - rukovanje

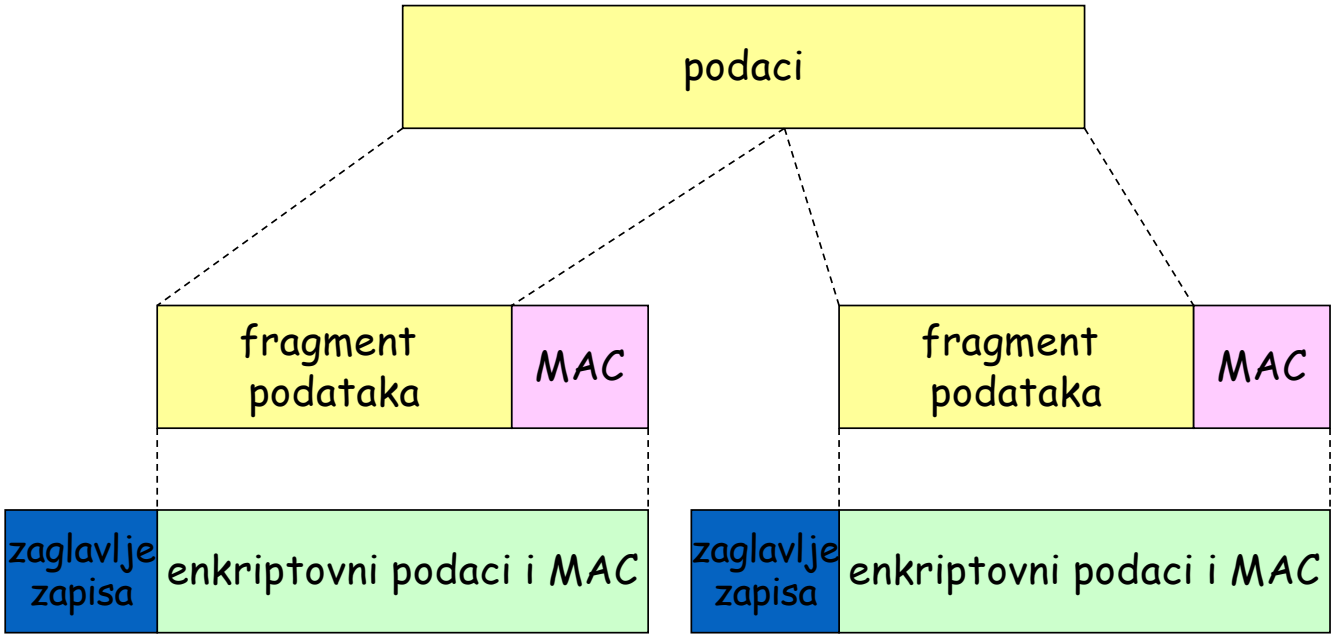
Poslednja 2 koraka su dodatna zaštita

- Klijent tipično nudi više algoritama, neki su jači neki slabiji
- Uljez može izbrisati jače algoritme iz liste
- Poslednja 2 koraka ovo sprečavaju
  - Poslednje dvije poruke su enkriptovane

Stvarni TLS - rukovanje

- Zbog čega dva random nonce-a?
- Pretpostavimo da Trudi snima sve poruke koje razmjenjuju Alisa i Bob
- Sledećeg dana Trudi uspostavlja TCP konekciju sa Bobom, šalje identičnu sekvencu zapisa
  - Bob (Amazon) misli da je Alis napravila dvije porudžbine iste stvari
  - Rešenje: Bob šalje različitu *random nonce* vrijednost za svaku konekciju. Samim tim će i ključevi za enkripciju biti različiti za svaku konekciju
  - Trudine poruke neće proći Bobovu provjeru integriteta

TLS record protocol



Zaglavlje zapisa: tip, verzija, dužina;

MAC: računa se na osnovu broja u sekvenici i MAC ključa

fragment: maksimalno 16 KB

TLC izvođenje ključeva

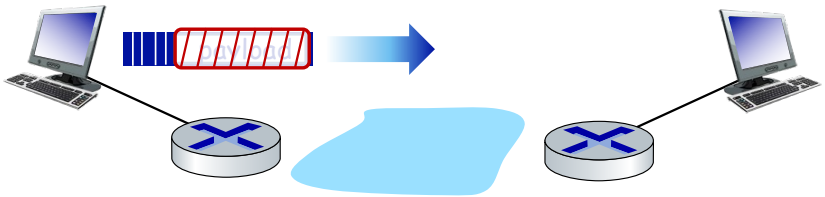
- Klijentov *nonce*, serverov *nonce* i *pre-master secret* su ulazni argumenti generatora pseudoslučajnih brojeva
  - Kreira se master ključ
- Master ključ i novi *nonce*-ovi se koriste kao ulaz drugog generatora random brojeva: dobija se "blok ključeva"
- Blok ključeva se dijeli na:
  - klijentski MAC ključ
  - serverki MAC ključ
  - Klijentski ključ enkripcije
  - Serverski ključ enkripcije
  - Vektor inicijalizacije klijenta (IV – *Initalization Vector*)
  - Vektor inicijalizacije servera (IV)

Bezbjednost računarskih mreža

- Šta je mrežna sigurnost?
- Principi kriptografije
- Integritet poruke, autentifikacija
- Sigurnost elektronske pošte
- Sigurnost TCP konekcija: TLS
- Sigurnost na mrežnom sloju: IPsec
- Sigurnost u bežičnim i mobilnim mrežama
- Sigurnost u praksi: firewall-i i IDS

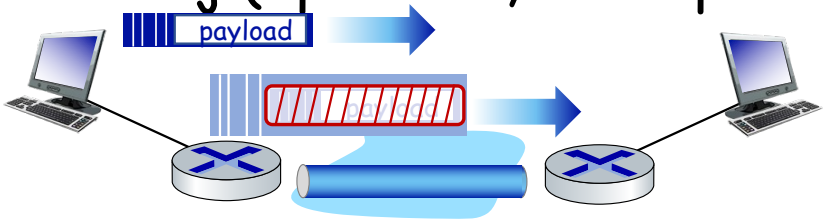
IP Sec

- Pruža enkripciju na nivou datagrama, autentifikaciju, integritet
  - Za korisnički i kontrolni saobraćaj (npr. BGP, DNS poruke)
- dva "moda":



Transportni mod:

- samo payload datagrama se enkriptuje i autentifikuje



Tunel mod:

- čitav datagram se enkriptuje, autentifikuje
- Enkriptovani datagram se enkapsulira u novi datagram sa novim IP zaglavljem, i tuneluje se do destinacije

Dva IPsec protokola

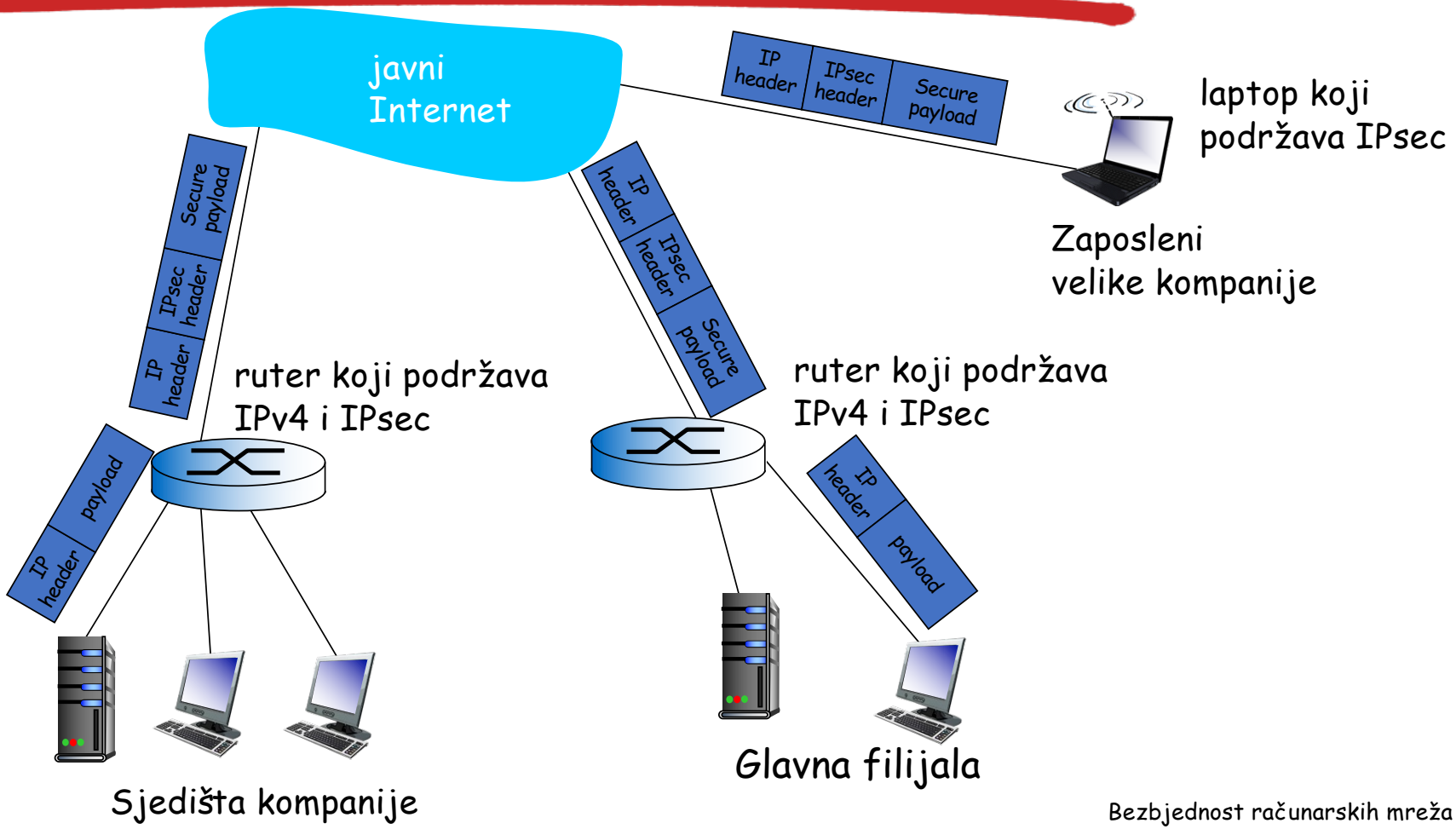
- *Authentication Header* (AH) protokol [RFC 4302]
  - Pruža autentifikaciju izvora i zaštitu integriteta podataka ali ne i tajnost
- *Encapsulation Security Protocol* (ESP) [RFC 4303]
  - Pruža autentifikaciju izvora, zaštitu integriteta podataka i tajnost
  - Više je zastupljen od AH

Virtuelne privatne mreže (VPN)

Motivacija:

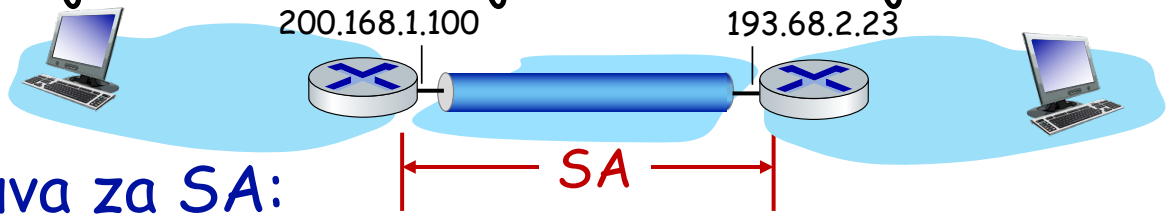
- Institucije često žele privatne mreže zbog sigurnosti komunikacije
  - skupo: odvojeni ruteru, linkovi, DNS infrastruktura.
- VPN: saobraćaj različitih sektora institucije šalje se preko javnog Interneta
  - Saobraćaj se enkriptuje prije slanja na javni Internet
  - Logički se odvaja od ostalog saobraćaja na Internetu

Virtuelne private mreže (VPN)



Sigurne asocijacije (SA)

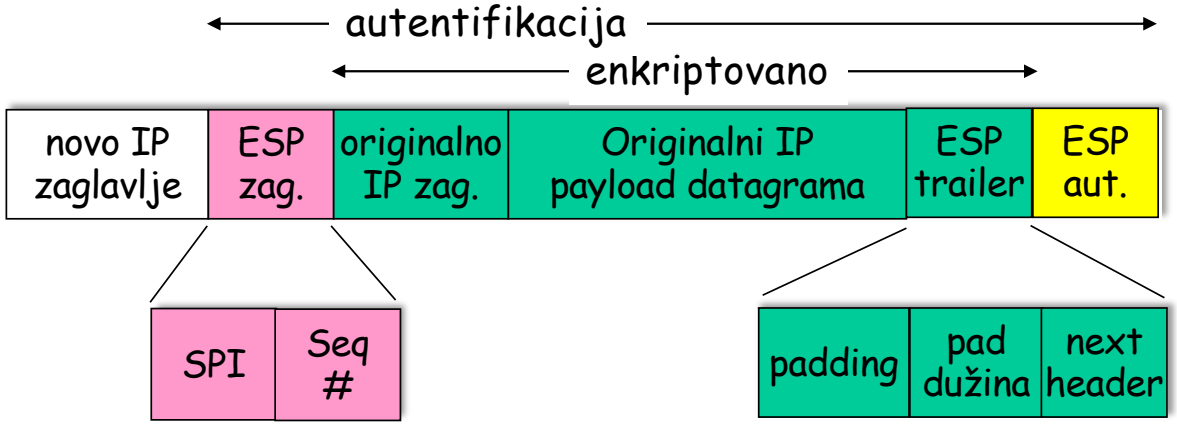
- Prije slanja podataka, uspostavlja se sigurna asocijacija (SA) između entiteta koji šalje i entiteta koji prima podatke (usmjerena)
- Po završetku prenosa, enititet koji je primao podatke održava informacije o stanju sigurne asocijacije.
  - Podsjećanje: TCP krajnji sistemi takođe održavaju informacije o stanju
  - IP nije konektivno orijentisan; IPsec je konektivno orijentisan!



R1 čuva za SA:

- 32-bitni identifikator: *Security Parameter Index* (SPI)
- izvorišni SA interfejs (200.168.1.100)
- destinacioni SA interfejs (193.68.2.23)
- tip korišćene enkripcije
- ključ za enkripciju
- tip provjere integriteta
- ključ za autentifikaciju

IPsec datagram



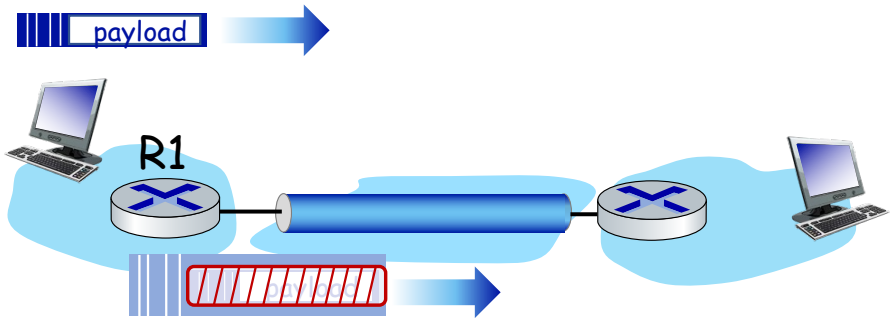
tunel mod ESP

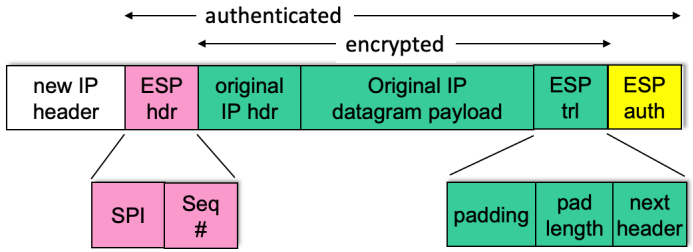
- ESP trailer: *padding* za blok šifratore
- ESP zaglavlje:
  - SPI, kako bi entitet koji prima podatake znao šta treba da uradi
  - Broj u sekvenci, u cilju prevencije replay napada
- MAC u ESP autentifikator polju se kreira pomoću dijeljenog tajnog ključa

ESP tunel mode: akcije

na R1:

- Dodaje ESP trailer originalnom datagramu (koji uključuje originalna polja zaglavlja!)
- Enkriptuje rezultat koristeći algoritam i ključ specificiran u SA
- Dodaje ESP zaglavlja na početak enkriptovanog sadržaja iz prethodnog koraka
- Kreira MAC kod koristeći algoritam i ključ specificiran u SA
- Dodaje MAC kreirajući novi payload
- Kreira novo IP zaglavlje, nova polja u zaglavlju, adresira datagram na kraj tunela





IPsec brojevi u sekvenci

- Za novu SA, pošiljalac inicijalizuje broj u sekvenci na 0
- Svaki put kada se datagram pošalje na SA:
  - Pošiljalac uvećava brojač sekvence
  - Postavlja novu vrijednost broj u sekveni u zaglavlje
- Cilj:
  - Spriječiti napadača da skenira i ponovi paket
  - Prijem dupliranih, autentifikovanih IP paketa može izazvati prekid servisa
- metod:
  - Destinacija provjerava da li je paket duplikat
  - Ne vodi evidenciju o svim primljenim paketima, već umejsto toga vodi računa o prozoru poslednje primljenih paketa

IPsec sigurnosna baza podataka

Security Policy Database (SPD)

- Polisa: predajni entitet treba da zna da li je potrebno koristiti IPsec za dati datagram,
- Polisa se čuva u bazi sigurnosnih polisa (Security Policy Database - SPD)
- Potrebno je da zna koji SA treba koristiti
  - Može koristiti: izvorišnu i destinacionu IP adresu, vrstu protokola

SPD: "kako" to uraditi

Security Assoc. Database (SAD)

- Krajnji entitet čuva stanje SA u bazi sigurnih asocijacija (Security Association Database - SAD)
- Kada šalje IPsec datagram, R1 pristupa SAD bazu da utvrdi kako treba obraditi paket
- Kada IPsec datagram stigne do R2, R2 ispituje SPI u IPsec datagramu, indeksira SAD sa SPI, obrađuje datagram na odgovarajući način

SAD: "šta" uraditi

Zaključak: IPsec servisi

Trudi osluškuje komunikaciju između R1 i R2; ne zna korišćene ključeve

- Da li će Trudi biti u stanju da vidi originalni sadržaj datagrama? Da li vidi izvorišnu i destinacionu IP adresu, transportni protokol, port za aplikaciju?
- Da li može da zamijeni bite a da se to ne može detektovati?
- Da li može da se maskira kao R1 koristeći IP adresu rutera R1?
- Da li može da ponovi datagram a da se to ne detektuje?

IKE: Internet Key Exchange

- Prethodni primjeri: manuelno uspostavljanje IPsec sigurnih asocijacija između krajnjih tačaka:
  - Primjer SA:
    - SPI: 12345
    - Izvorišna IP: 200.168.1.100
    - Dest. IP: 193.68.2.23
    - Protokol: ESP
    - Algoritam enkripcije: 3DES-cbc
    - HMAC algoritam: MD5
    - Ključ za enkripciju: 0x7aeaca…
    - HMAC ključ:0xc0291f…
- Manuelno setovanje ključeva je nepraktično za VPN sa nekoliko stotina krajnjih tačaka
- Umjesto toga koristi se IPsec IKE (*Internet Key Exchange*)

IKE: PSK i PKI

- Autentifikacija (dokaži ko si) ili sa
  - Unaprijed dijeljenim ključem (*Pre-Shared Key*) ili
  - sa PKI (javni/privatni ključevi i sertifikati).
- PSK: obije strane startuju sa istim tajnim ključem
  - Izvršavaju IKE da se međusobno autentifikuju i da generišu IPsec sigurne asocijacije (po jednu u svakom smjeru), uključujuću enkripciju, ključeve za autentifikaciju
- PKI: obije strane startuju sa parom javnog i privatnog ključa i odgovarajućeg sertifikata
  - Izvršavaju IKE da bi se međusobno autentifikovali, dobili IPsec sigurne asocijacije (po jedna u svakom smjeru).
  - Slično rukovanju kod TLS-a.

IKE faze

- IKE ima dvije faze
  - faza 1: uspostavlja se bidirekciona IKE SA
    - napomena: IKE SA se razlikuje od IPsec SA
    - poznato kao ISAKMP sigurna asocijacija
  - faza 2: ISAKMP se koristi za sigurno pregovaranje para IPsec sigurnih asocijacija
- faza 1 ima dva moda: agresivni mod i glavni mod
  - Agresivni mod koristi manje poruka
  - Glavni mod pruža zaštitu identiteta i fleksibilniji je

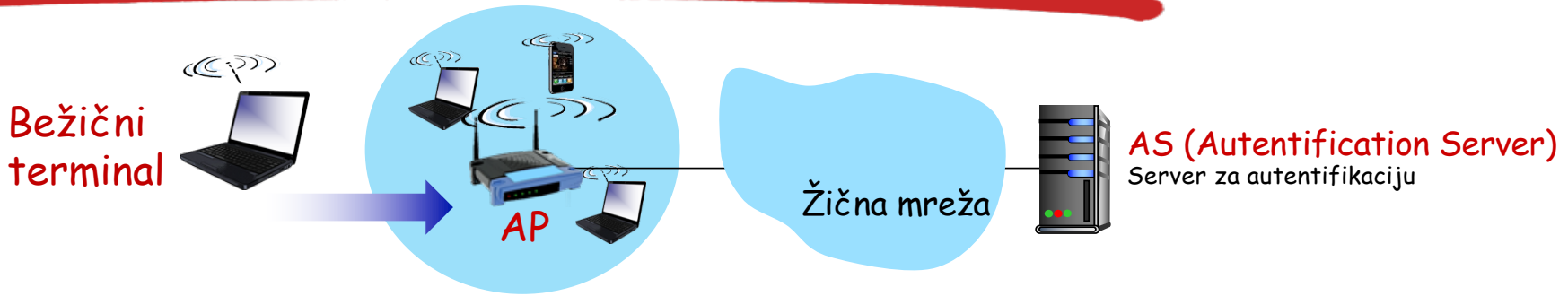
IPsec rezime

- IKE razmjena poruka za pregovaranje algoritama, tajnih ključeva, SPI brojeva
- ili AH ili ESP protokol (ili oba)
  - AH pruža zaštitu integriteta i omogućava autentifikaciju izvora
  - ESP protokol (sa AH) dodatno pruža enkripciju
- IPsec peer-ovi mogu biti dva krajnja sistema, dva rutera/firewall-a ili ruter/firewall i krajnji sistem

Bezbjednost računarskih mreža

- Šta je mrežna sigurnost?
- Principi kriptografije
- Integritet poruke, autentifikacija
- Sigurnost elektronske pošte
- Sigurnost TCP konekcija: TLS
- Sigurnost na mrežnom sloju: IPsec
- Sigurnost u bežičnim mrežama
- Sigurnost u praksi: firewall-i i IDS

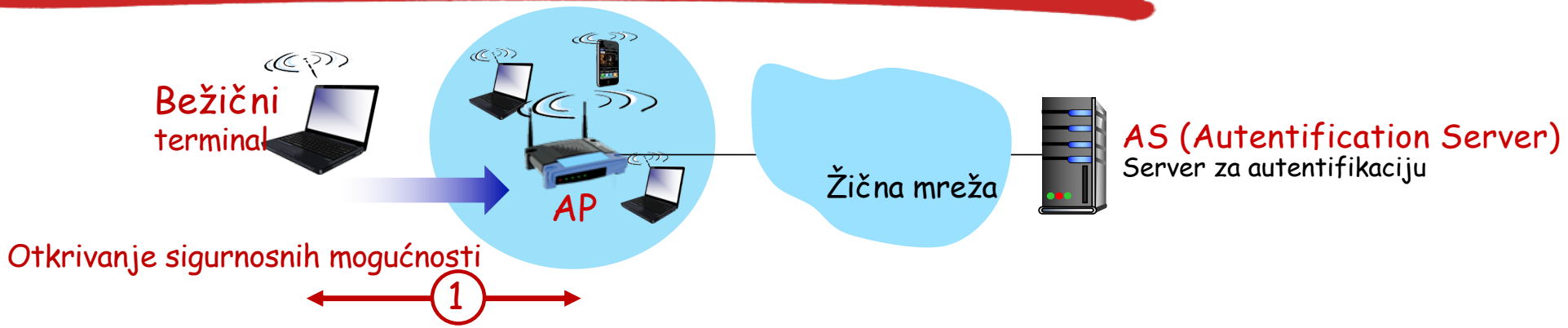
802.11: autentifikacija, enkripcija



Bežični terminal mora da se:

- poveže na access point: upostavlja komunikaciju preko bežičnog linka
- autentifikuje

802.11: autentifikacija, enkripcija

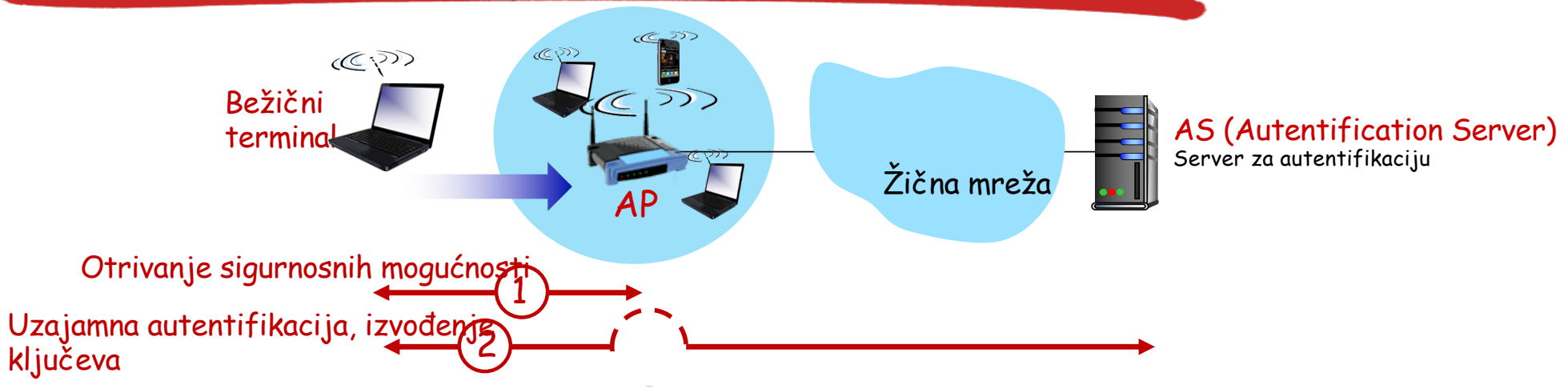


1 Otkrivanje sigurnosnih mogućnosti:

- AP oglašava svoje prisustvo, forme autentifikacije i enkripcije
- Uređaj zahtijeva specifičnu formu autentifikacije, enkripcije

Iako mobilni terminal i AP već razmjenjuju poruke, uređaji nisu još autentifikovani niti su usaglasili ključeve za enkripciju

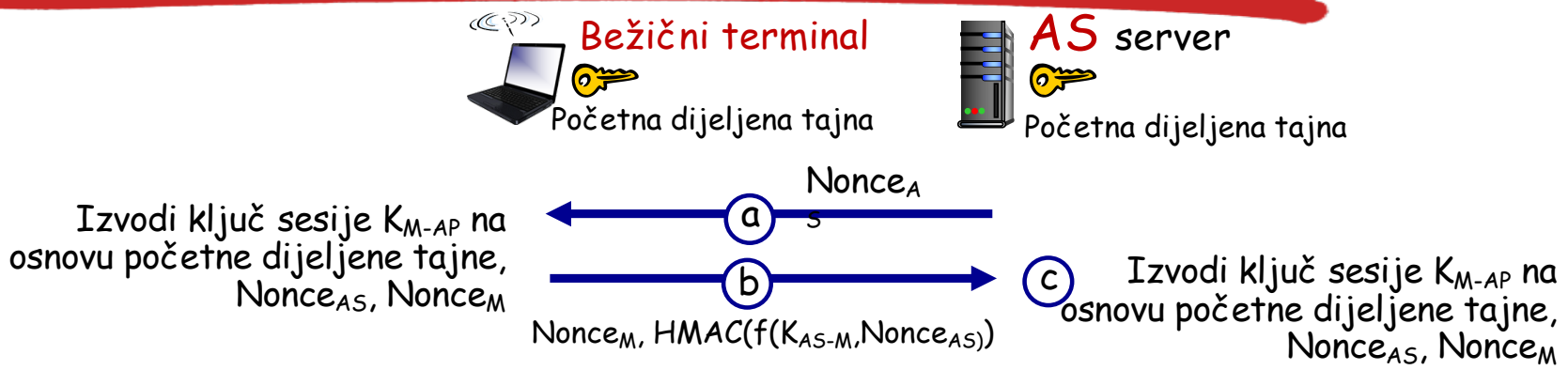
802.11: autentifikacija, enkripcija



(2) Uzajamna autentifikacija i izvođenje dijeljenog ključa za simetričnu enkripciju:

- AS i terminal već imaju zajedničku tajnu informaciju (npr. password)
- AS i terminal koriste dijeljenju tajnu, nonce vrijednosti (prevencija napada ponavljanjem), MAC kodove bazirane na kriptografskom heširanju i MAC adrese da se uzajmano autentifikuju
- AS i terminal generišu dijeljeni (simetrični) ključ sesije

802.11: WPA3 rukovanje

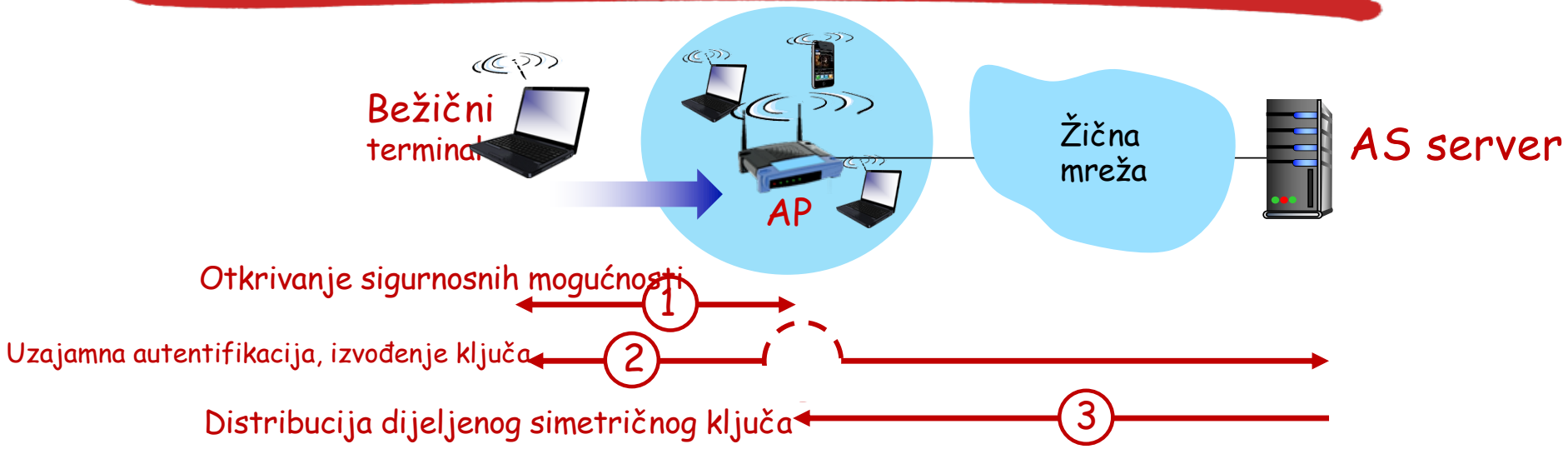


ⓐ AS generiše $Nonce_{AS}$, šalje ga bežičnom terminalu

ⓑ Mobilni terminal prima $Nonce_{AS}$
- generiše $Nonce_M$
- generiše simentrični ključ sesije K_{M-AP} koristeći $Nonce_{AS}$, $Nonce_M$, inicijalnu dijeljenu tajnu, svoju MAC adresu i MAC adresu AS servera
- šalje $Nonce_M$, i HMAC-om potpisuje vrijednost $Nonce_{AS}$ koristeći početnu dijeljenu tajnu

ⓒ AS generiše simetrični ključ sesije K_{M-AP}

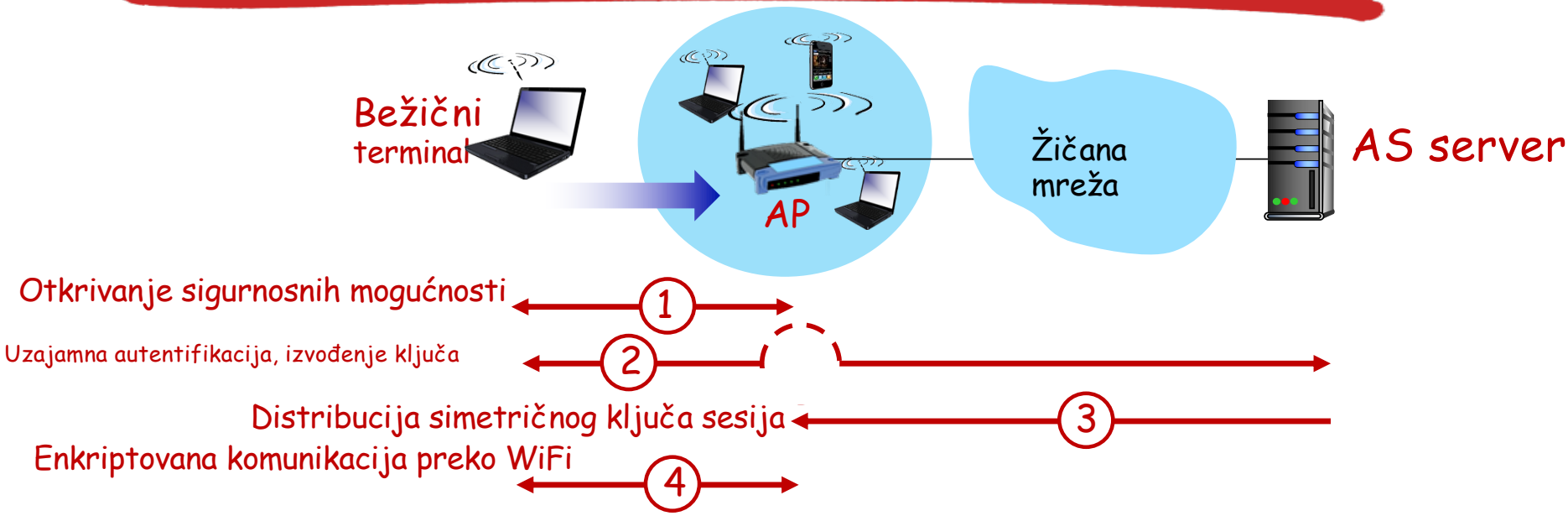
802.11: autentifikacija, enkripcija



③ Distribucija dijeljenog simetričnog ključa sesije (npr. za AES enkripciju)

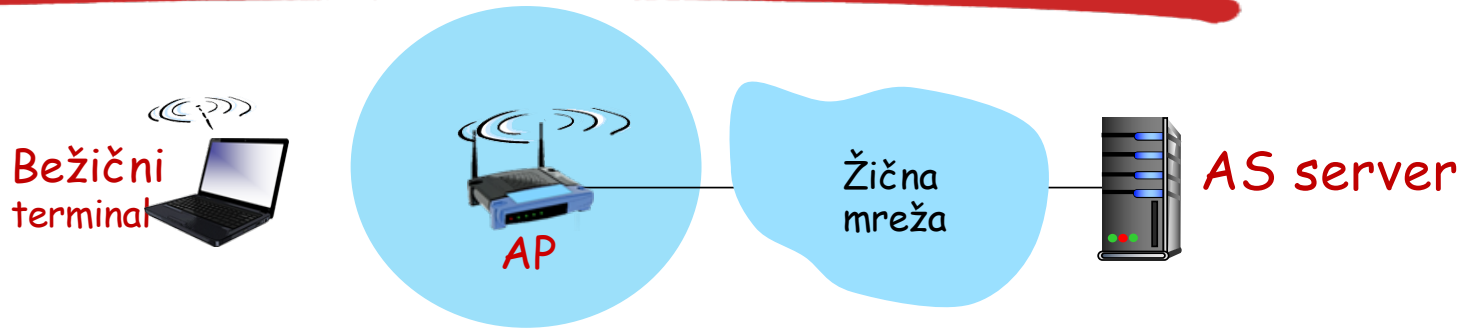
- Isti ključ je generisan od stane bežičnog terminala i AS-a
- AS informiše AP o dijeljenom simetričnom ključu

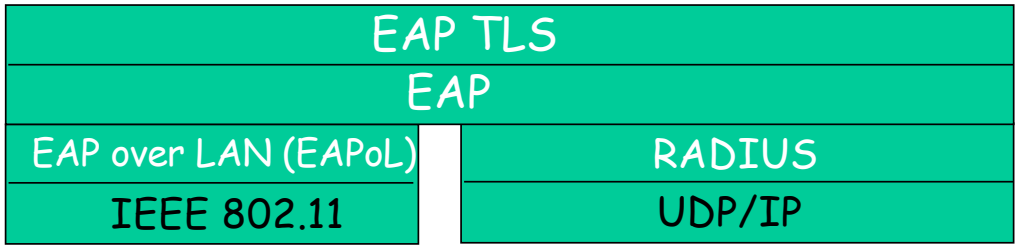
802.11: autentifikacija, enkripcija



④ Enkriptovana komunikacija između bežičnog terminala i udaljenog hosta preko AP

802.11: autentifikacija, enkripcija





Extensible Authentication Protocol (EAP) [RFC 3748] definiše end-to-end protokol komunikacije između mobilnih uređaja i AS-a baziran na zahtjev/odgovor modelu.

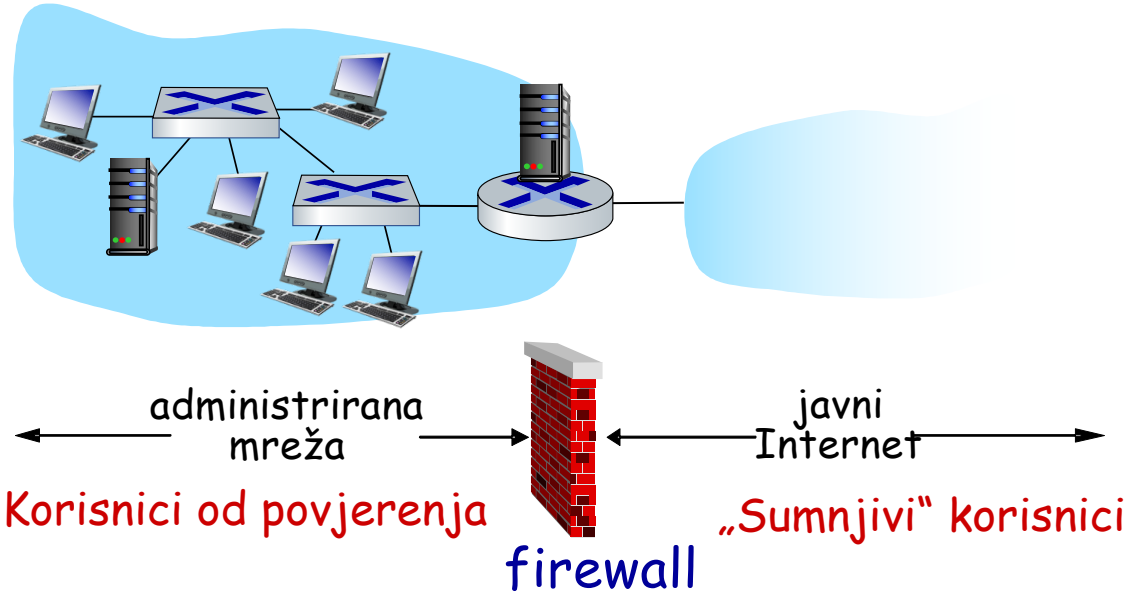
Bezbjednost računarskih mreža

- Šta je mrežna sigurnost?
- Principi kriptografije
- Integritet poruke, autentifikacija
- Sigurnost elektronske pošte
- Sigurnost TCP konekcija: TLS
- Sigurnost na mrežnom sloju: IPsec
- Sigurnost u bežičnim mrežama
- Sigurnost u praksi: firewall-i i IDS

Firewall

Firewall

Izoluje internu mrežu organizacije od Interneta, dozvoljavajući ulaz određenim paketima a blokirajući ostale



Firewall: Zašto?

Sprečavaju DoS napade:

- SYN flooding: napadač uspostavlja veliki broj lažnih TCP konekcija tako da meta napada nema dovoljno resursa za prihvatanje „stvarne" TCP konekcije

Sprečava neovlašćen pristup internim podacima i njihovu modfikaciju

- Npr. napadač mijenja home stranicu nekog sajta

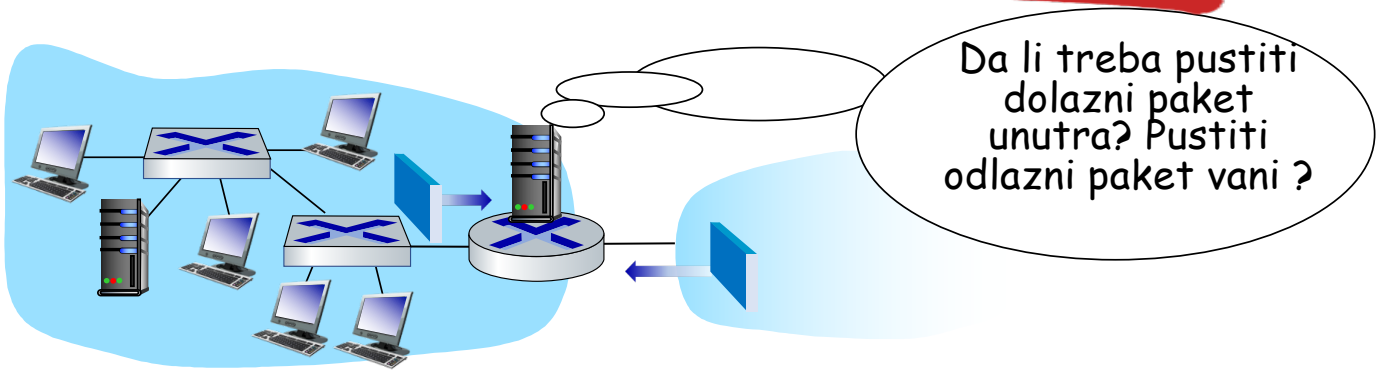
Dozvoljava samo autorizovan pristup internoj mreži

- Set autentifikovanih korisnika/hostova.

Tri tipa firewall-a:

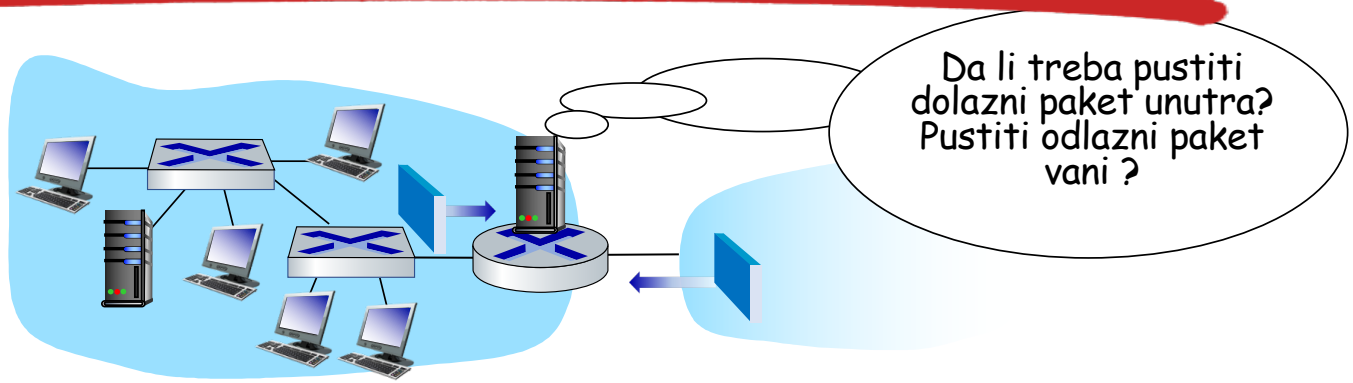
- Stateless filteri paketa
- Stateful filteri paketa
- Aplikacioni gejtveji

Stateless filtriranje paketa



- Interna mreža povezana na Internet preko ruter firewall-a
- Filtriranje se vrši paket-po-paket, odluka o prosleđivanju/odbacivanju paketa bazira se na:
  - Izvorišnoj IP adresi, destinacionoj IP adresi
  - TCP/UDP izvorišnim, destinacionim brojevima porta
  - ICMP tipu poruke
  - TCP SYN, ACK bitima

Stateless filtriranje paket



- Primjer 1: blokiraj dolazne i odlazne datagrame sa IP protokol poljem = 17 i sa izvorišnim ili destinacionim brojem porta = 23
  - Rezultat: svi dolazni i odlazni UDP tokovi telnet konekcije biće blokirani
- Primjer 2: blokiraj dolazne TCP segmente sa ACK=0
  - Rezultat: sprečava eksterne klijente da uspostave konekcije prema internim klijentima, ali dozvoljava internim klijentima da uspostave konekcije prema uređajima van interne mreže

Stateless filtriranje paket: više primjera

Polisa	Podešavanje firewall-a
Spriječiti pristup Web stranicama van mreže	Odbaci sve odlazne pakete sa destinacionim brojem porta 80, prema bilo kojoj IP adresi
Bez dolaznih TCP konekcija, izuzev onih usmjerenih na javni Web server institucije	Odbaci sve dolazne TCP SYN segmente prema bilo kojoj IP adresi izuzev 130.207.244.203, port 80
Spriječiti da audio saobraćaj potisne ostali saobraćaj u mreži	Odbaci sve dolazne UDP pakete – izuzev DNS paketa i broadcast paketa rutera.
Spriječiti zloupotrebu mreže za realizaciju *smurf* DoS napada.	Odbaci sve ICMP paketa usmjerene na „broadcast" adresu (npr. 130.207.255.255)
Onemogućiti vidljivost mreže *traceroute* alatima	Odbaciti sve odlazne ICMP TTL expired pakete

Access Control Lists

ACL: tabela pravila koja se primjenjuju od vrha ka dnu nad dolaznim paketima: (akcija, uslov) parovi: slično OpenFlow tabeli prosleđivanja!

akcija	Izvorišna adresa	Dest. adresa	Protokol	Izvorišni port	Dest. port	flag bit
dozvoli	222.22/16	izuzev 222.22/16	TCP	> 1023	80	sve
dozvoli	izuzev 222.22/16	222.22/16	TCP	80	> 1023	ACK
dozvoli	222.22/16	izuzev 222.22/16	UDP	> 1023	53	---
dozvoli	izuzev 222.22/16	222.22/16	UDP	53	> 1023	----
odbaci	sve	sve	sve	sve	sve	sve

Stateful filtriranje paketa

- stateless filter paketa: alat težak za upotrebu
  - Prihvata paketa koji "nemaju smisla," npr. dest. port = 80, ACK bit 1, iako nema uspostavljenih konekcija:

akcija	Izvorišna adresa	Dest. adresa	Protokol	Izvorišni port	Dest. port	flag bit
dozvoli	izuzev 222.22/16	222.22/16	TCP	80	> 1023	ACK

- stateful filter paketa: prati status svake TCP konekcije
  - Prati uspostavljanje konekcije (SYN), raskidanje (FIN): na osnovu ovog određuje da li dolazni i odlazni paketi "imaju smisla"
  - Definiše timeout na firewall-u za konekcije koje nisu aktivne: paketi ovih konekcija neće biti prihvatani posle timeout-a

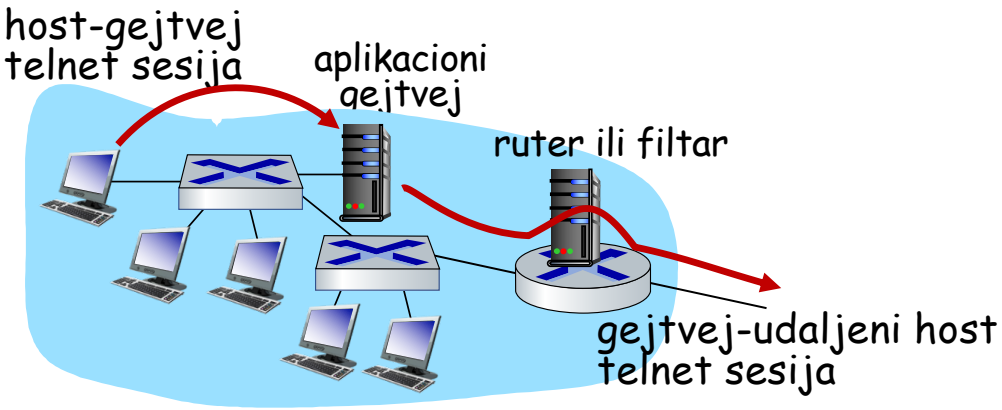
Stateful filtriranje paketa

ACL proširena tako da indicira potrebu za provjerom stanja konekcije prije prihvatanja paketa

akcija	Izvorišna adresa	Dest. adresa	Protokol	Izvorišni port	Dest. port	flag bit	Provjeri konekciju
dozvoli	222.22/16	izuzev 222.22/16	TCP	> 1023	80	sve	
dozvoli	izuzev 222.22/16	222.22/16	TCP	80	> 1023	ACK	X
dozvoli	222.22/16	izuzev 222.22/16	UDP	> 1023	53	---	
dozvoli	izuzev 222.22/16	222.22/16	UDP	53	> 1023	----	X
odbaci	sve	sve	sve	sve	sve	sve	

Aplikacioni gejtveji

- Filtriranje paketa na osnovu podataka iz određene aplikacije.
- Uzima u obzir identitet internih korisnika
- Primjer: dozvoliti određenim internim korisnicima telnet komunikaciju prema spolja



1. Zahtijeva da svi telnet korisnici upostavljaju telnet konekcije preko gejtveja
2. Za autorizovane korisnike, gejtvej će uspostaviti konekciju sa destinacionim hostom
   - gejtvej ima ulogu releja između dvije konekcije
3. Filter na ruteru blokira sve telnet konekcije koje nisi inicirane od strane gejtveja

Ograničenja firewall-a, gejtveja

- IP spoofing: ruter ne može znati da li podaci stvarno stižu od određenog izvora
- Ukoliko više aplikacija zahtijeva specijalni tretman, svaka od njih mora imati svoj aplikacioni gejtvej
- Klijentski softver mora znati kako da kontaktira gejtvej
  - Npr. mora setovati IP adresu proxy-a u Web pretraživaču
- filteri često koriste polise tipa sve ili ništa za UDP
- Kompromis: određeni stepen komunikacije sa spoljnim svijetom, uz određeni nivo sigurnosti
- Mnogi zaštićeni sajtovi i dalje su podložni napadima

Sistemi za detekciju upada

- Filtriranje paketa:
  - Analiziraju se samo TCP/IP zaglavlja
  - Ne provjerava se korelacija sesija
- *IDS: intrusion detection system*
  - *Deep packet inspection*: provjerava sadržaj paketa (npr. provjerava stringove u paketu i poredi ih sa bazom poznatih virusa, stringovima napada)
  - Ispituje se korelacija paketa
    - Skeniranje portova
    - Mapiranje mreže
    - DoS napad

Sistemi za detekciju upada

Višestruki IDS sistemi: različiti tipovi provjere na različitim lokacijama