Protokoli za autentifikaciju

Sadržaj

- Jednostavni sigurnosni protokoli
- Protokoli za autentifikaciju
- Autentifikacija i TCP
- Sigurnosni protokoli u praksi

Jednostavni sigurnosni protokoli

Protokoli.

- Protokoli – pravila ponašanja pri interakciji više subjekata.
  - Primer: postavljanje pitanja na času.
- Mrežni protokoli – pravila koja se primenjuju u mrežnim komunikacionim sistemima.
  - Primeri: HTTP, FTP, itd.
- Sigurnosni protokoli – pravila (komunikacije) u aplikacijama koje treba da obezbede siguran prenos informacija.
  - Primeri: SSL, IPSec, Kerberos, itd.

Jednostavni sigurnosni protokoli

Protokoli.

- Protokoli mogu da budu veoma ranjivi u pogledu sigurnosti.
- Veoma male (naizgled beznačajne) izmene protokola, mogu uneti velike promene u pogledu sigurnosti.
- Nekoliko dobro poznatih sigurnosnih protokola imaju ozbiljne slabosti.
  - Primer: GSM, WEP, itd.
- I u slučajevima kada protokoli nemaju slabosti, one se mogu pojaviti zbog loše implementacije.

Jednostavni sigurnosni protokoli

Idealni sigurnosni protokoli.

- Zadovoljavaju zahteve sigurnosti.
  - Zahtevi moraju da budu precizni.
- Efikasni.
  - Minimizovati računsku složenost – posebno kod zahtevnih sistema sa javnim ključem.
  - Minimizovati kašnjenja i potrebe za propusnim opsegom.
- Robusni.
  - Mora da radi i kada napadač pokušava da ga “razbije”.
  - Mora da radi i kada se menjaju uslovi prenosa.
- Lak za upotrebu i implementaciju, fleksibilan, ...
- Pokazuje se da je teško sve zadovoljiti!

Jednostavni sigurnosni protokoli

Ulazak u prostorije NSA.

- Razmotrimo protokol za ulazak u bezbednosno ranjivo okruženje kao što je NSA.
- Zaposleni imaju ID karticu, koju moraju stalno da nose.
- Da bi ušli u zgradu:
  1. Postavljaju ID karticu u čitač.
  2. Unose PIN.
  3. PIN je ispravan?
     - Da? Ulaze.
     - Ne? Intervencija obezbeđenja.

Jednostavni sigurnosni protokoli

Podizanje novca sa bankomata.

1. Postavljaja se kartica u čitač bankomata.
2. Unosi se PIN.
3. PIN je ispravan?
   - Da? Omogućiti transakcije.
   - Ne? Blokiranje ili oduzimanje kartice.

Jednostavni sigurnosni protokoli

Protokoli u vojnim primenama.

- Identifikacija prijatelja, odnosno neprijatelja – *Identify Friend or Foe* (IFF).
  - Sprečavanje napada na sopstvene snage i minimizacija greške ne-napadanja neprijatelja.
- Primer:
  - Snage Južne Afrike (*South African Armed Forces* – SAAF) su stacionirane u Namibiji.
  - Bore se protiv neprijatelja na teritoriji Angole (koriste avione tipa MIG).
  - Kada radar detektuje avion SAAF kako mogu da budu sigurni čiji je?
  - Radar:
    - šalje slučaj broj *N*,
    - SAAF šifruje *N* sa ključem *K* i vraća *E*(*N*, *K*).
  - Ključ K je tajna!

Jednostavni sigurnosni protokoli

IFF protokol.

MIG

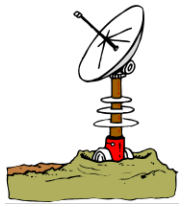
Angola

Namibija

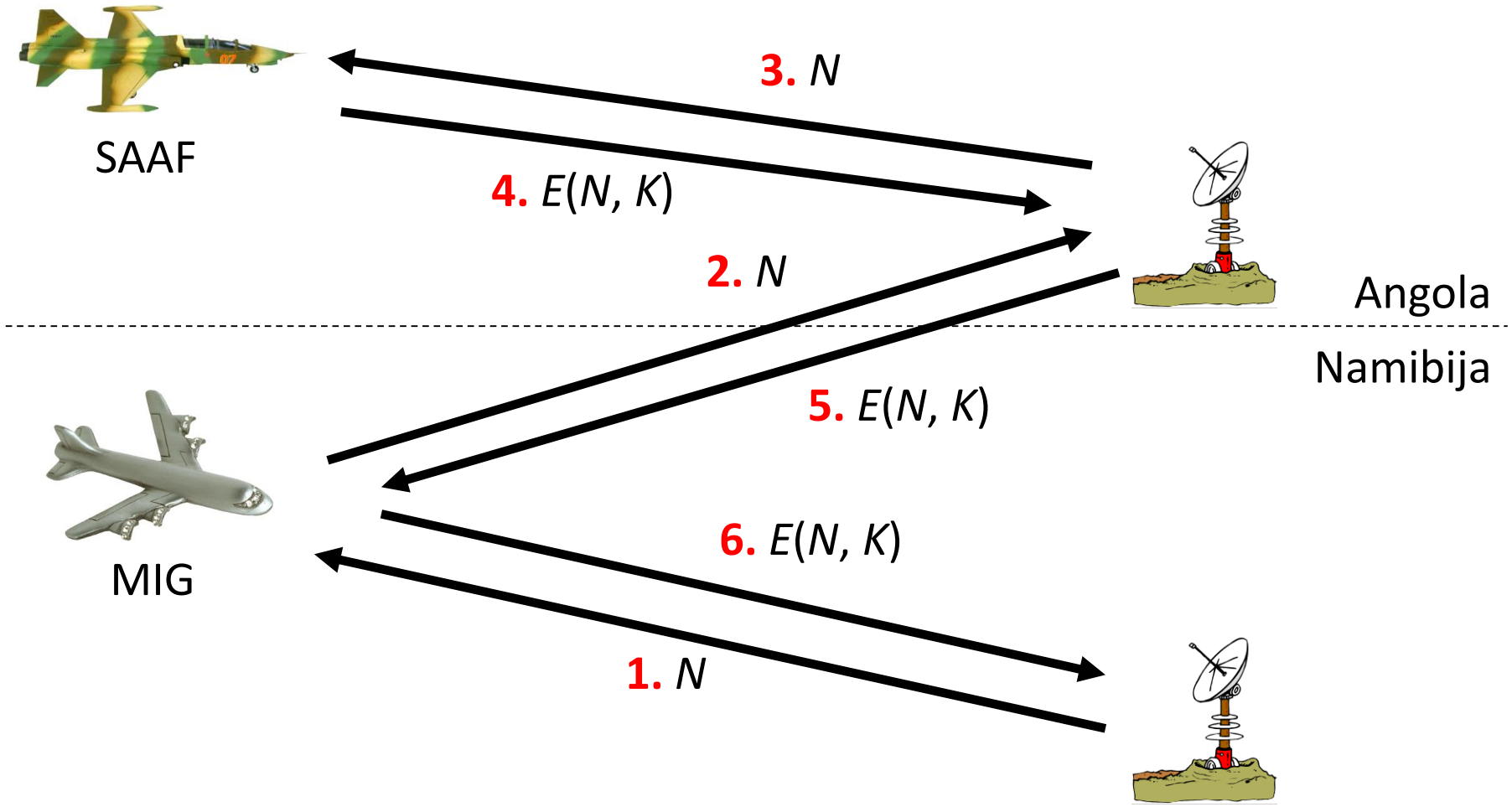
SAAF

2. $E(N, K)$

1. N



***MIG in the Middle* (*Man in the Middle*).**



Protokoli za autentifikaciju

Autentifikacija.

- Alisa treba da dokaže svoj identitet Bobu.
  - Alisa i Bob mogu da budu osobe ili računari.
- Može se zahtevati i da Bob dokaže svoj identitet Alisi (uzajmna autentifikacija).
  - Da li koristiti isti protokol za obe strane?
- Često se koristi simetrični (sesijski) ključ.
  - Poverljivost i/ili integritet.
- U određenim okolnostima, mogu se zahtevati:
  - upotreba samo javnog ključa,
  - upotreba samo simetričnog ključa,
  - upotreba samo heš funkcija,
  - ...

Protokoli za autentifikaciju

Autentifikacija.

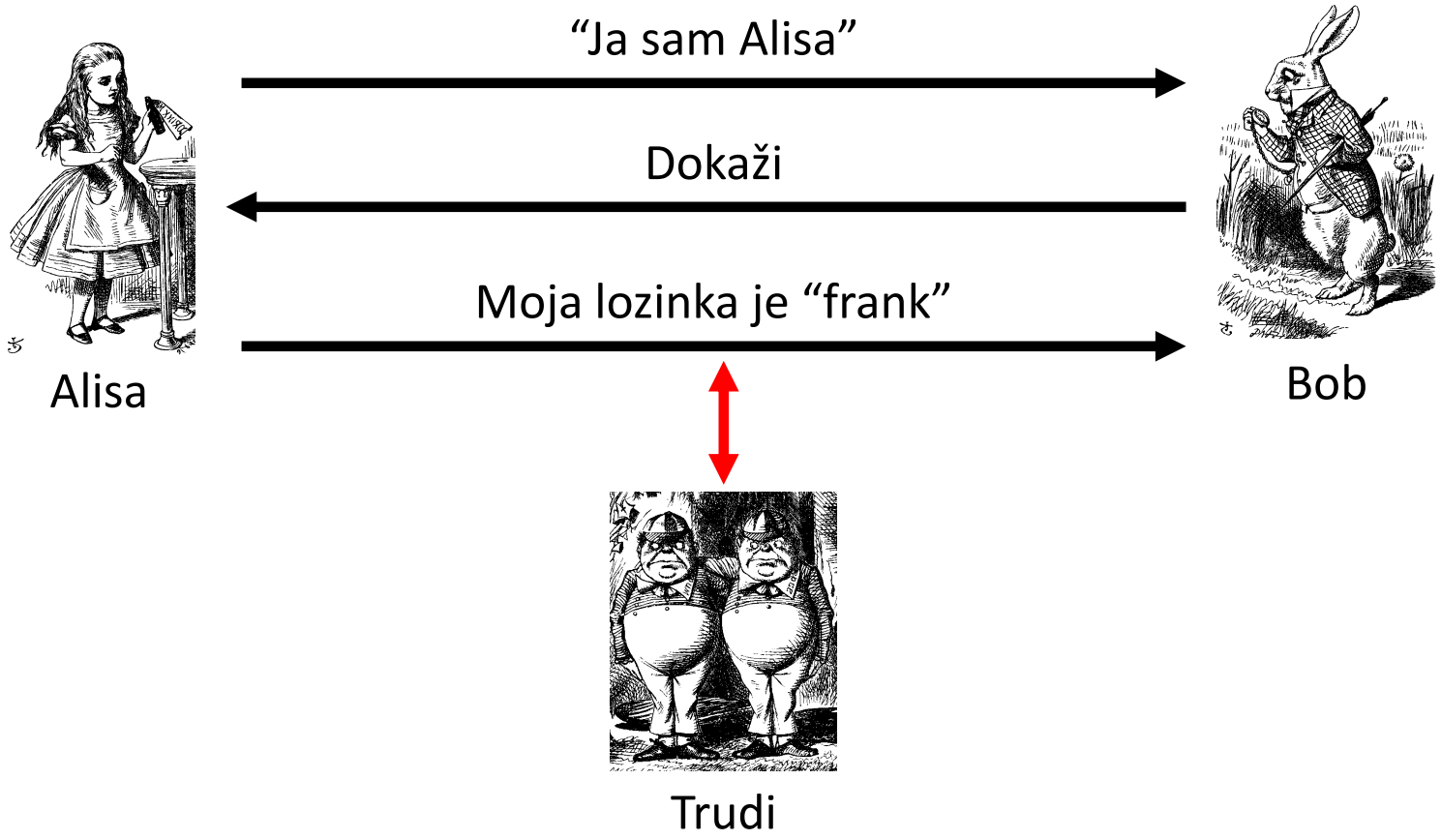
- Postupak autentifikacije na računar koji nije u mreži je relativno jednostavan.
  - Glavna slabost je napad na softver za autentifikaciju (kasnije objašnjeno).
- Autentifikacija preko mreže je mnogo složeniji problem.
  - Napadač može pasivno da analizira saobraćaj.
  - Napadač može da ponovi poruke (aktivan napad).
  - Mogući su i drugi aktivni napadi (kreiranje novih poruka, brisanje, promena sadržaja, itd.)

Protokoli za autentifikaciju

Jednostavna autentifikacija.

- Jednostavno i vrlo uslovno “OK” za računare koji nisu u mreži.
- Nije dovoljno sigurno za računare u mreži.
  - Moguć je napad ponavljanjem poruka.
  - Bob mora da zna Alisinu lozinku.
  - Lozinka se šalje kao otvoreni tekst.

Napad na autentifikacioni protokol.



Napad na autentifikacioni protokol.

- Ovo je napad ponavljanjem poruka.
- Problem je još ozbiljniji ako Alisa koristi istu lozinku na više mesta.
- Kako se može predupredi̇ti?

Protokoli za autentifikaciju

Upotreba heša.

- Prednost: ne šalje se lozinka.
  - Bob ne mora da je zna, ali mora da zna heš lozinke.
  - Trudi ne može da je sazna.
- Međutim, još uvek može da bude predmet napada.
  - Napad ponavljanjem poruke je izvodljiv.

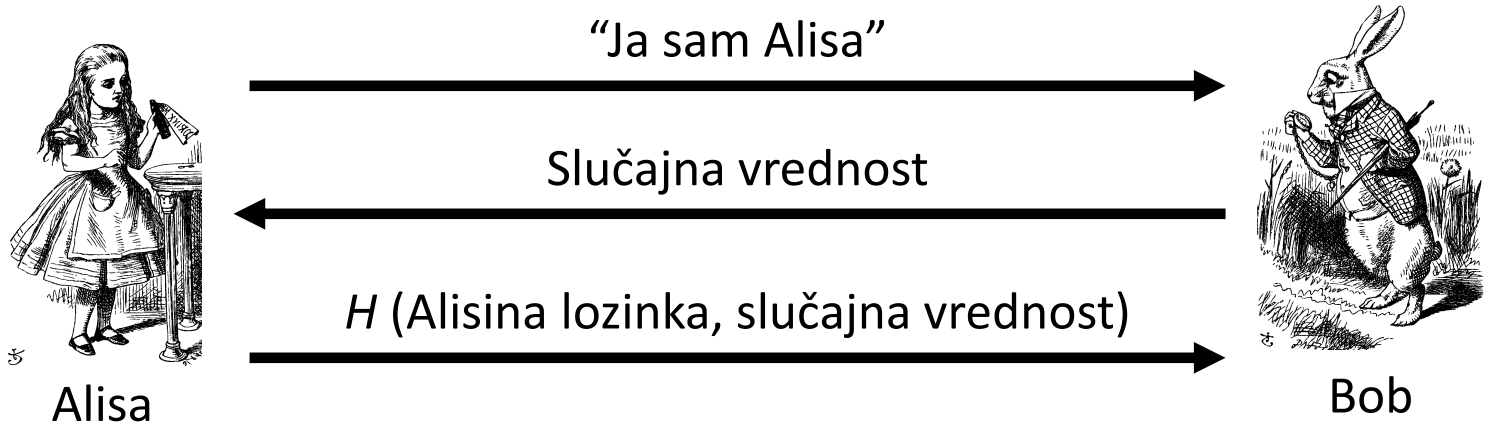
Protokoli za autentifikaciju

Izazov-odgovor (*challenge-response*).

- Da bi se sprečio napad ponavljanja poruke koristi se metod *challenge-response*.
- Neka Bob želi da autentifikuje Alisu.
  - Bob šalje poruku, takozvani izazov (*challenge*) Alisi.
  - Samo Alisa može da generiše ispravan odgovor (*response*).
  - *Challenge* se bira tako da je nemoguć napad ponavljanjem poruke.
- Kako se to postiže?
  - Lozinka je nešto što (bi trebalo da) zna samo Alisa.
  - Challenge je slučajan sadržaj koji se ne ponavlja (*number used once – nonce*).
  - Alisa na osnovu primljene poruke i lozinke, po dogovorenom algoritmu generiše odgovor.

Protokoli za autentifikaciju

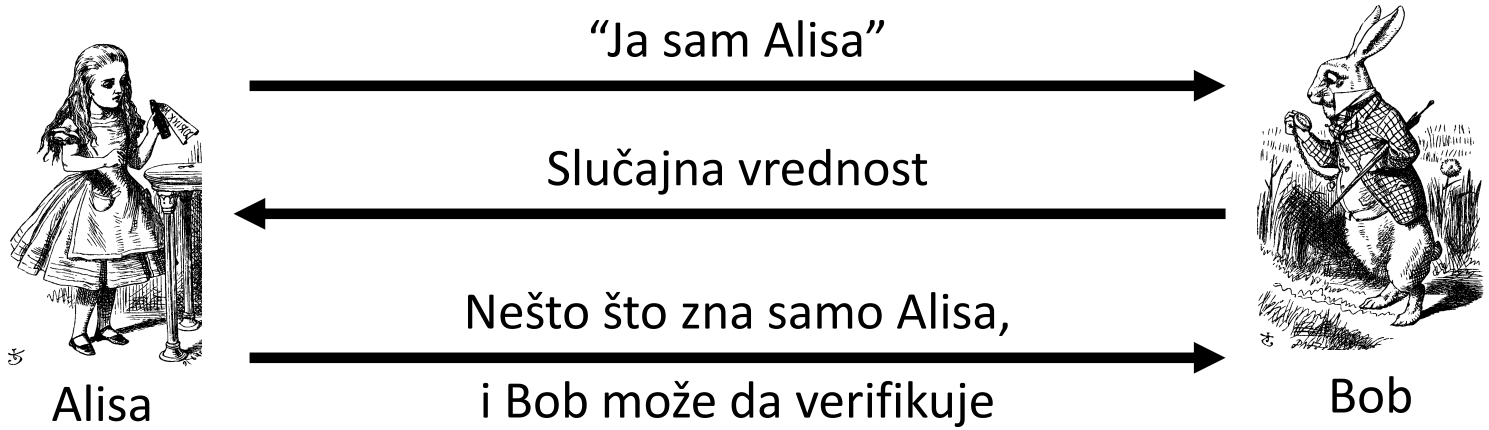
Izazov-odgovor (*challenge-response*).



- Slučajna vrednost je *challenge*.
- Heš izračunat na osnovu Alisine lozinke i slučajne vrednosti je *response*.
- Slučajna vrednost sprečava napad ponavljanja poruke.
- Napomena: u ovom scenariju Bob mora da zna Alisinu lozinku.

Protokoli za autentifikaciju

Upotreba simetrične kriptografije.



- Heš se pokazao kao korisno rešenje.
- Mogu li se na neki način iskoristiti simetrični kriptosistemi?

Upotreba simetrične kriptografije.

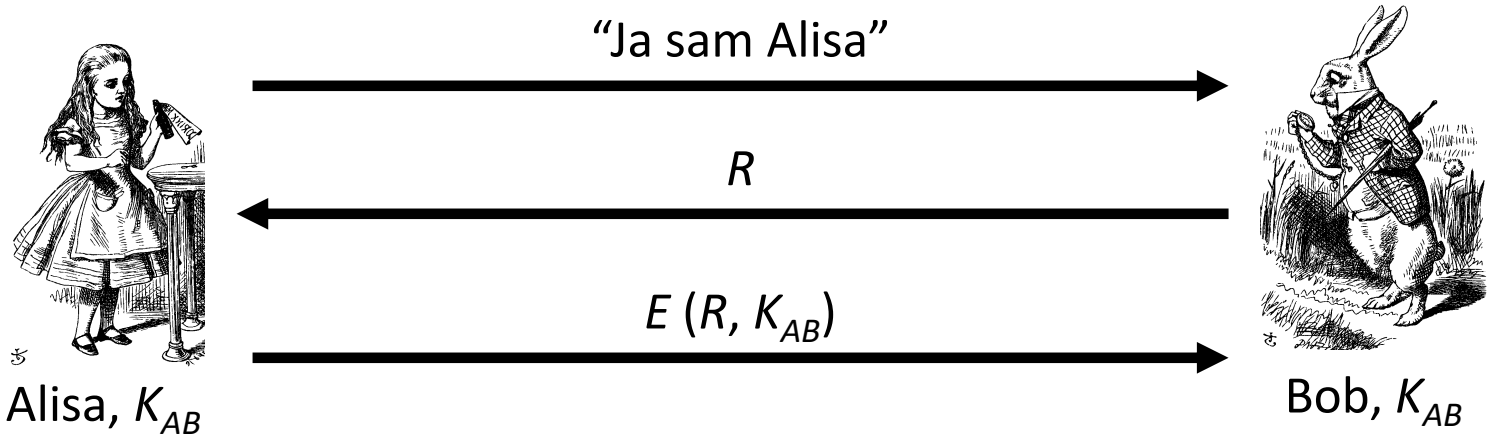
- Označavanje.
  - Šifrovanje otvorenog teksta *P* ključem *K*: $C = E(P, K)$.
  - Dešifrovanje šifrata *C* ključem *K*: $P = D(C, K)$.
  - Razmatraju se samo napadi na protokole, ne i napadi na kriptološka rešenja.
    - Podrazumeva se da je kriptološki algoritam siguran.

Protokoli za autentifikaciju

Autentifikacija sa simetričnim ključem.

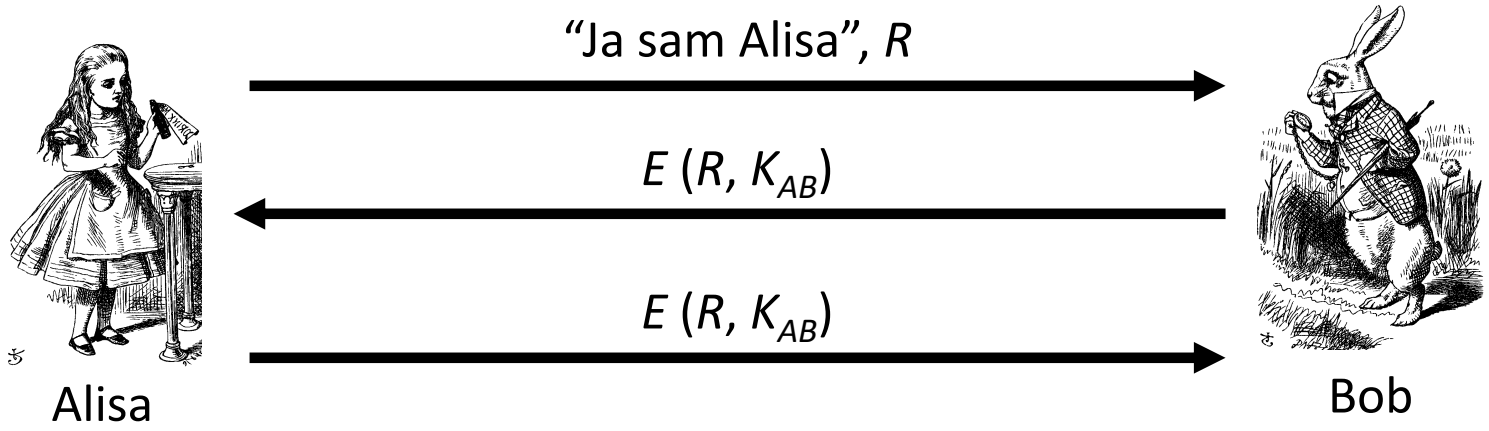
- Alisa i Bob dele simetrični ključ K_{AB}.
  - Ključ K_{AB} znaju samo Alisa i Bob.
- Autentifikacija se zasniva na posedovanju znanja o deljenoj tajni (deljenom simetričnom ključu).
- Kako to može da se postigne?
  - Trudi ne sme da sazna vrednost ključa.
  - Mora se sprečiti napad ponovljenog slanja.

Autentifikacija sa simetričnim ključem.



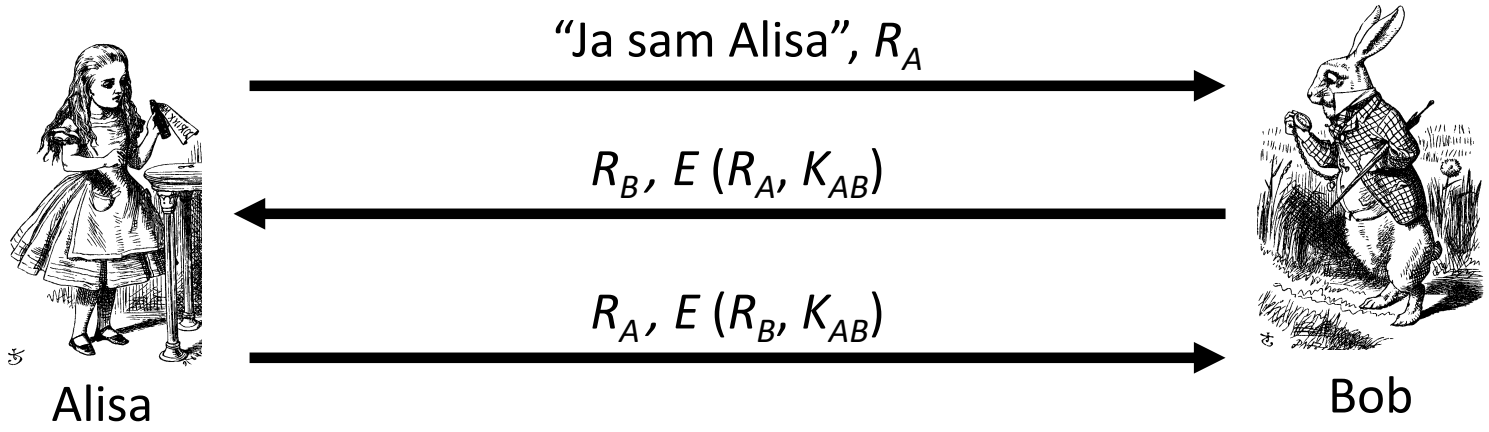
- Sličan *challenge-response* protokolu.
- Alisa šifruje slučajnu vrednost R simetričnim ključem K_{AB}.
- Siguran način da Bob autentifikuje Alisu, međutim, Alisa neće autentifikovati Boba.
- Kako možemo da postignemo da se Alisa i Bob uzajamno autentifikuju?

Uzajamna autentifikacija sa simetričnim ključem.



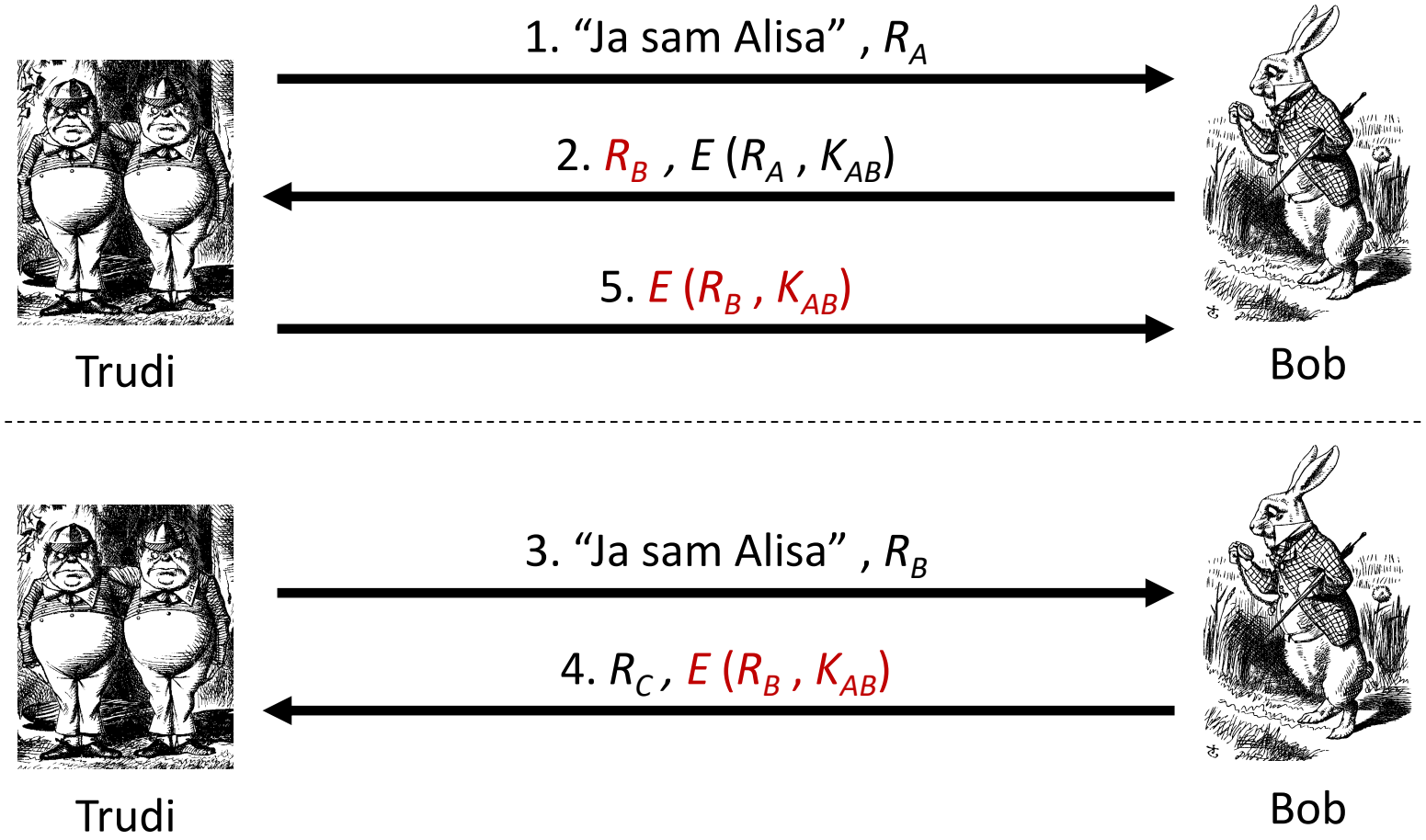
- Šta je pogrešno kod ovog pristupa?
- Alisa bi mogla da bude Trudi (ili bilo ko drugi).

Uzajamna autentifikacija sa simetričnim ključem.



- Ovo izgleda kao očigledan način za uzajamnu autentifikaciju.
- Međutim, ono što je očigledno ne mora uvek da bude i dobro.

Napad na uzajamnu autentifikaciju sa simetričnim ključem.

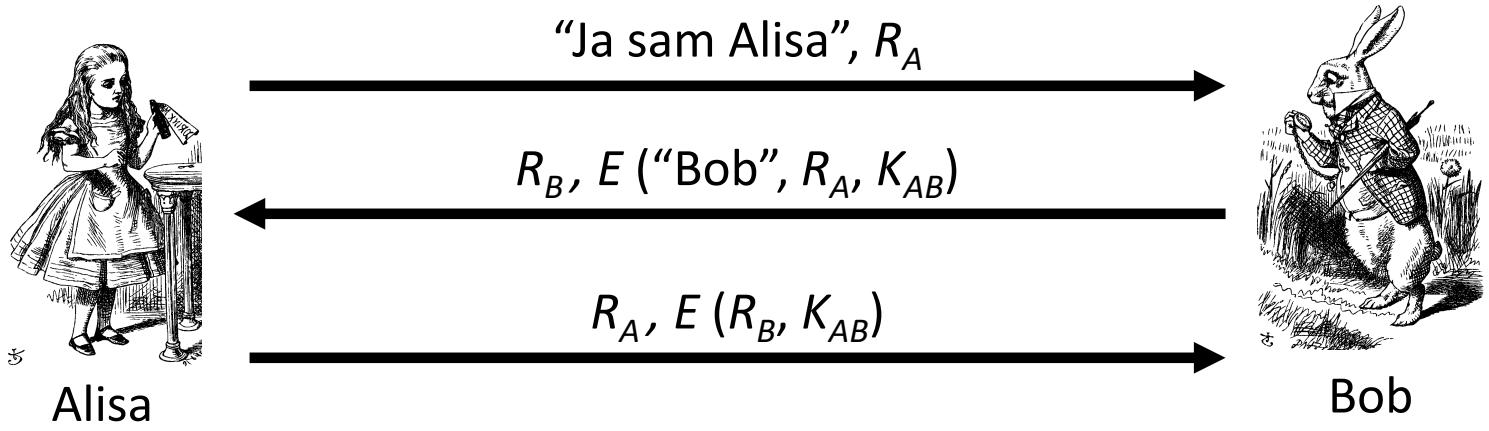


Protokoli za autentifikaciju

Pitanje sigurnosti.

- Jednosmerni protokoli za autentifikaciju nisu sigurni za uzajamnu autentifikaciju.
- Protokoli kriju mnoge zamke!
- “Očigledna” rešenja ne noraju uvek biti sigurna.
- Ako se “nešto” u protokolu izmeni, može se narušiti sigurnost.
  - Ovo su česti propusti!
  - Primer: Internet protokoli.

Uzajamna autentifikacija sa simetričnim ključem.

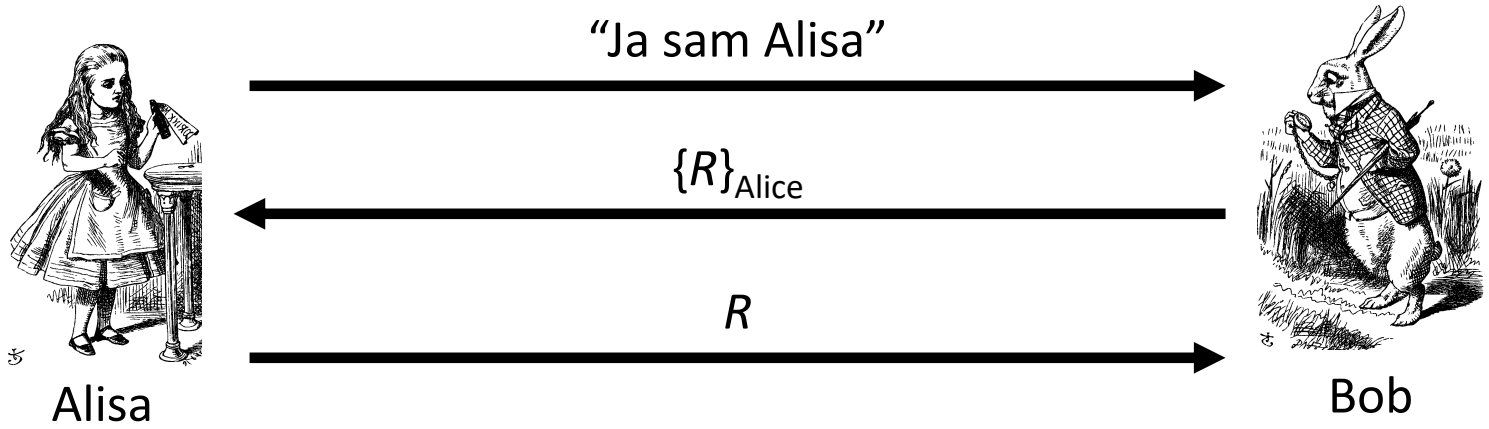


- Uključićemo korisnički identitet u postupak šifrovanja.
- Da li ova "mala" promena može da pomogne?
- Odgovor je: da!

Upotreba kriptografije sa javnim ključem.

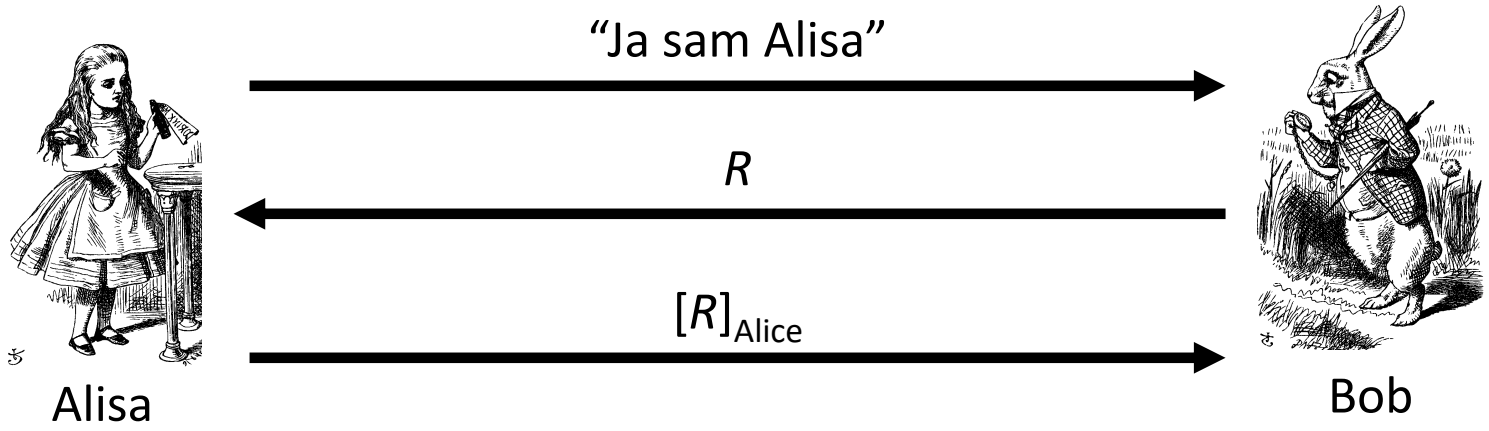
- Označavanje.
  - Šifrovanje poruke *M* Alisinim javnim ključem: $\{M\}_{Alice}$.
  - Digitalno potpisivanje poruke *M* Alisinim privatnim ključem: $[M]_{Alice}$.
  - Onda važi:
    - $[\ \{M\}_{Alice}\ ]_{Alice} = M$
    - $\{\ [M]_{Alice}\ \}_{Alice} = M$
  - Bilo ko može da obavi operacije sa javnim ključem.
  - Samo Alisa može da koristi njen privatni ključ (digitalno potpisivanje).

Autentifikacija sa javnim ključem.



- Da li je ovaj protokol siguran?
- Trudi je snimila šifrovanu poruku koju je Bob poslao Alisi: $C=\{M\}_{Alice}$.
- Trudi može da pošalje C Alisi.
- Alisa će je dešifrovati i poslati otvoreni tekst!

Autentifikacija sa javnim ključem.



- Da li je ovaj protokol siguran?
- Trudi može da "natera" Alisu da potpiše bilo šta!
- Rešenje: ne treba koristiti isti par ključeva za potpisivanje i šifrovanje.

Protokoli za autentifikaciju

Upotreba kriptografije sa javnim ključem.

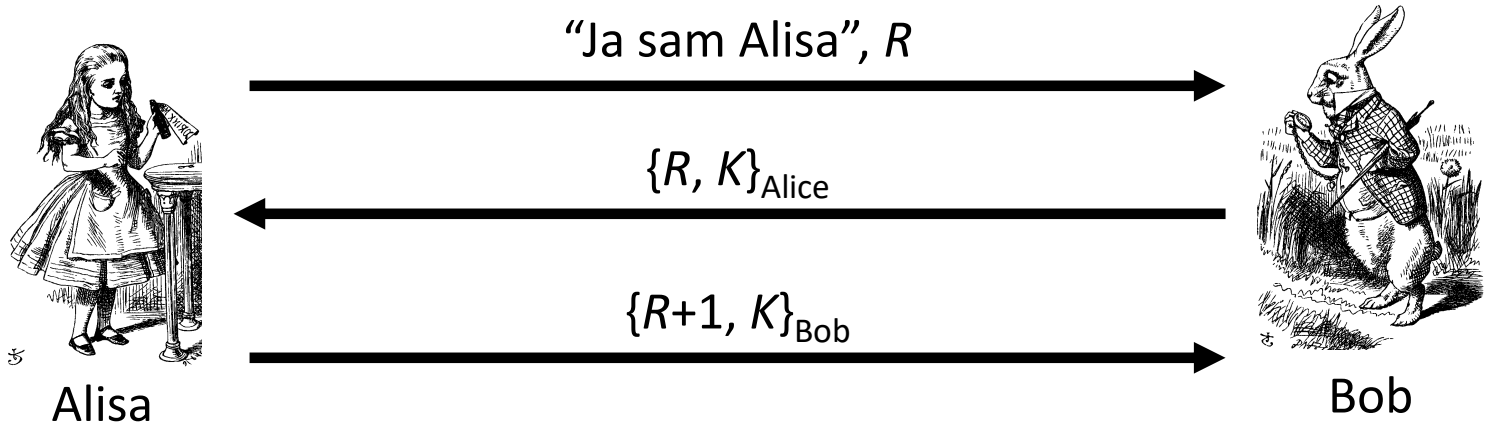
- Nikada ne treba koristiti isti par ključeva za šifrovanje i potpisivanje.
- Jedan par ključeva se koristi za šifrovanje i dešifrovanje.
- Drugi par ključeva se koristi za potpisivanje i verifikaciju potpisa.

Protokoli za autentifikaciju

Sesijski ključ.

- Nakon procesa autentifikacije, veoma često je potrebna razmena sesijskog (simetričnog) ključa.
  - Sesijski ključ je simetrični ključ za jednu komunikaciju.
  - Ključ se koristi kako bi se obezbedila poverljivost.
  - Može da se koristi i za obezbeđivanje integriteta.
- Može li se u okviru protokola za autentifikaciju razmeniti simetrični ključ?

Autentifikacija i razmena sesijskog ključa.



- Da li je ovaj protokol siguran?
- Deluje OK za razmenu ključa, ali nema uzajamne autentifikacije!
- Može li se modifikovati protokol tako da se umesti šifrovanja koristi digitalno potpisivanje?

Autentifikacija i razmena sesijskog ključa.

- Da li je ovaj protokol siguran?
- Postignuta uzajamna autentifikacija i sigurna razmena sesijskog ključa!
- Dakle, protokol deluje OK.

Autentifikacija i razmena sesijskog ključa.

- Da li je ovaj protokol siguran?
- Bilo ko može da vidi $\{R, K\}_{Alice}$ i $\{R+1, K\}_{Bob}$, ali to, u opštem slučaju, ne umanjuje sigurnost.
- Dakle, protokol deluje OK.

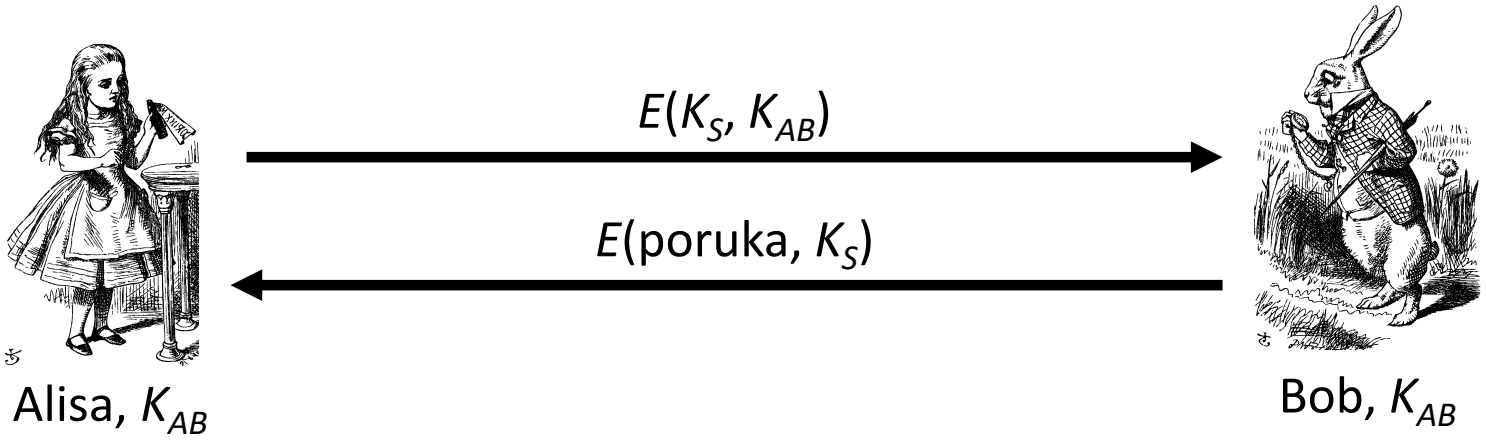
***Perfect Forward Secrecy* (PFS)**.

- Problem:
  - Alisa šifruje poruku sa (razmenjenim) simetričnim ključem K_{AB} i pošalje šifrat Bobu.
  - Trudi snimi šifrat i kasnije izvrši uspešan napad na Alisin ili Bobov računar i pronađe K_{AB}.
  - Trudi može naknadno da dešifruje sve što je snimila!
- Cilj PFS:
  - Trudi ne može da dešifruje snimljeni šifrat, čak i u slučaju da dođe u podsed ključa K_{AB}.
- Da li je moguće realizovati PFS?

***Perfect Forward Secrecy* (PFS)**.

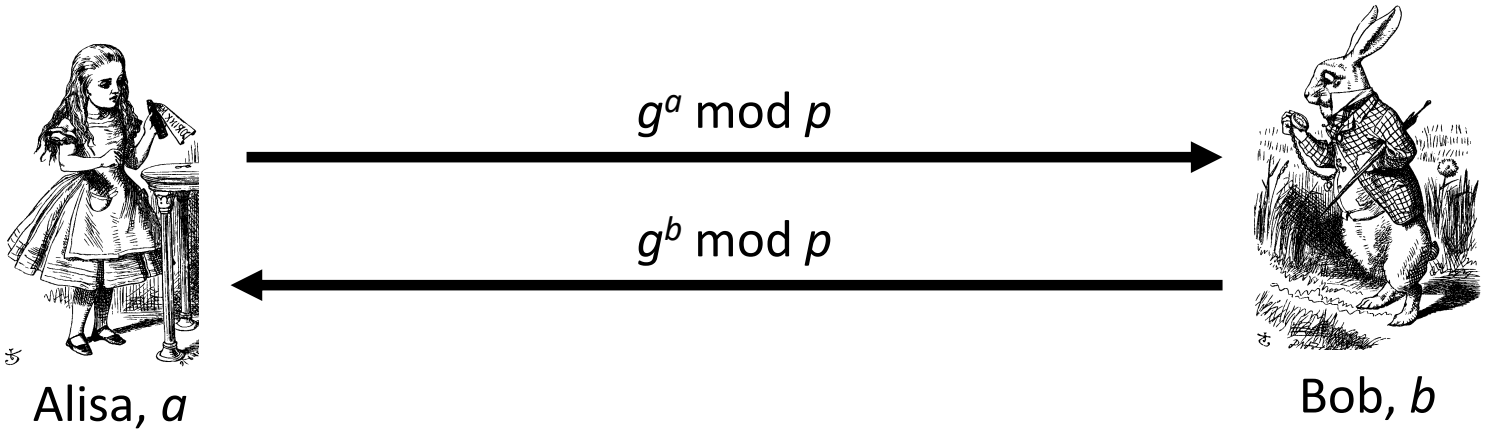
- Neka Alisa i Bob razmene simetrični ključ K_{AB}.
- Alisa i Bob neće koristiti K_{AB} za šifrovanje.
- Koristiće sesijski ključ K_S i uništiti ga nakon upotrebe.
- Problem:
  - Kako Alisa i Bob mogu da se dogovore oko izbora sesijskog ključa K_S, a da pritom ne naruše PFS?

Loš protokol za razmenu sesijskog ključa.



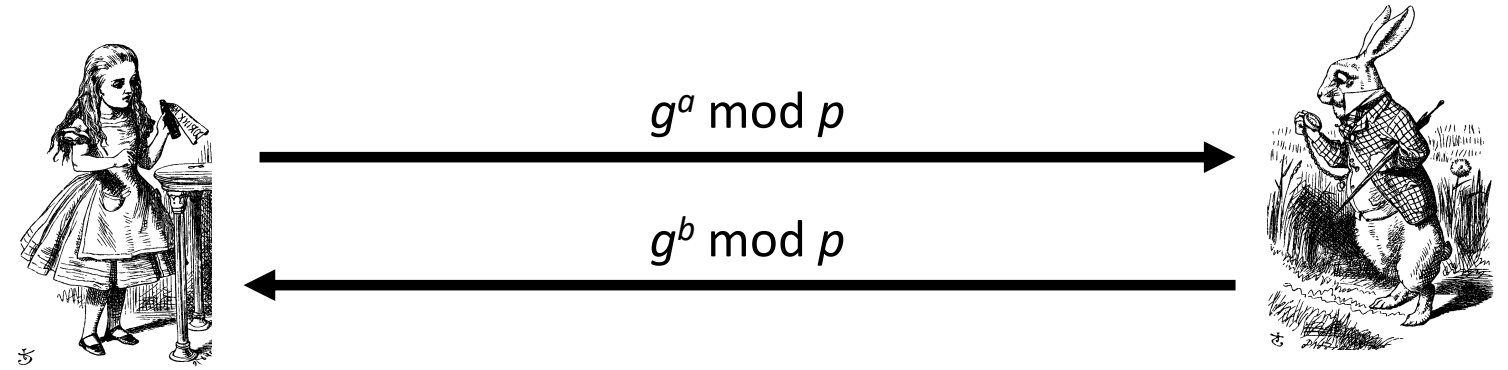
- Trudi može da snimi $E(K_S, K_{AB})$
- Ako trudi sazna K_{AB}, onda zna i K_S.

Upotreba Diffie-Helmanovog protokola za razmenu ključeva.



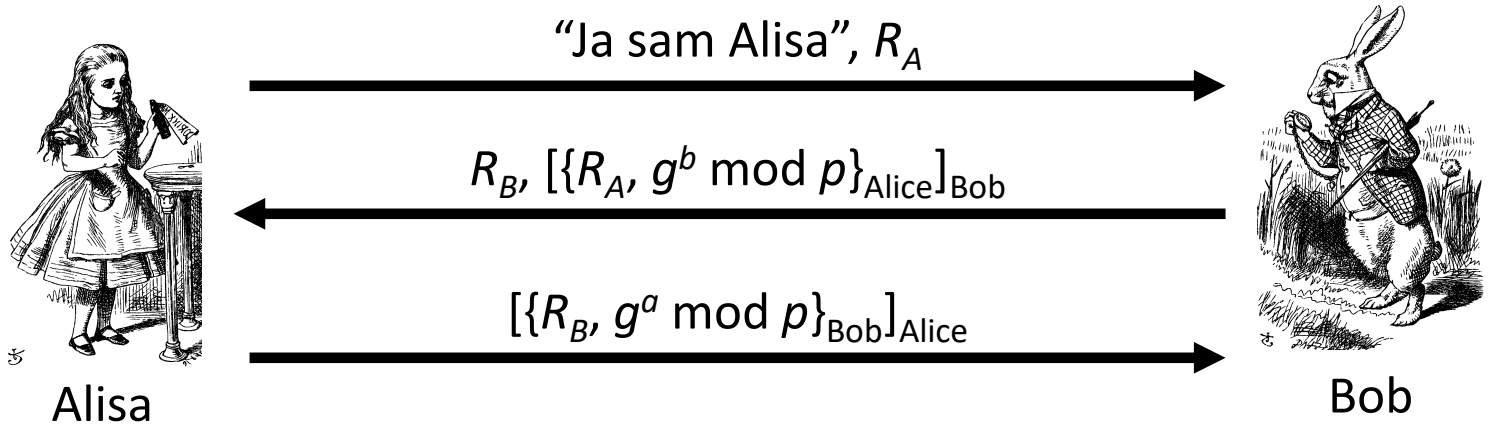
- Podsetnik: vrednosti g i p su javne.
- Međutim, Diffie-Hellman je osetljiv na napad tipa “čovek u sredini”.
- Kako postići PFS i sprečiti ovaj napad?

Upotreba Diffie-Helmanovog protokola za razmenu ključeva.



- Sesijski ključ je $K_S = g^{ab} \bmod p$.
- Sesijski ključ se koristi jednokratno.
- Alisa zaboravi vrednost a, Bob zaboravi vrednost b.
- Kasnije ni Alisa ni Bob ne mogu da izračunaju K_S – naravno da ni Trudi nema veću šansu.

Uzajamna autentifikacija, razmena sesijskog ključa i PFS.



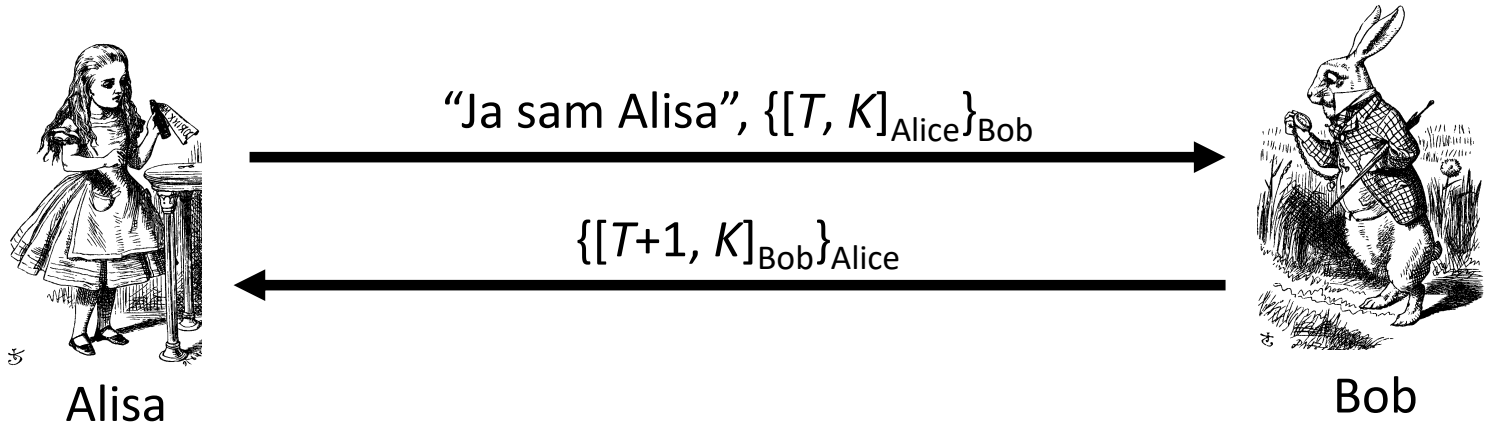
- Sesijski ključ je $K_S = g^{ab} \bmod p$.
- Sesijski ključ se koristi jednokratno.
- Alisa zaboravi vrednost a, Bob zaboravi vrednost b.
- Trudi nije u stanju da rekonstruiše sesijski ključ.

Protokoli za autentifikaciju

Timestamp.

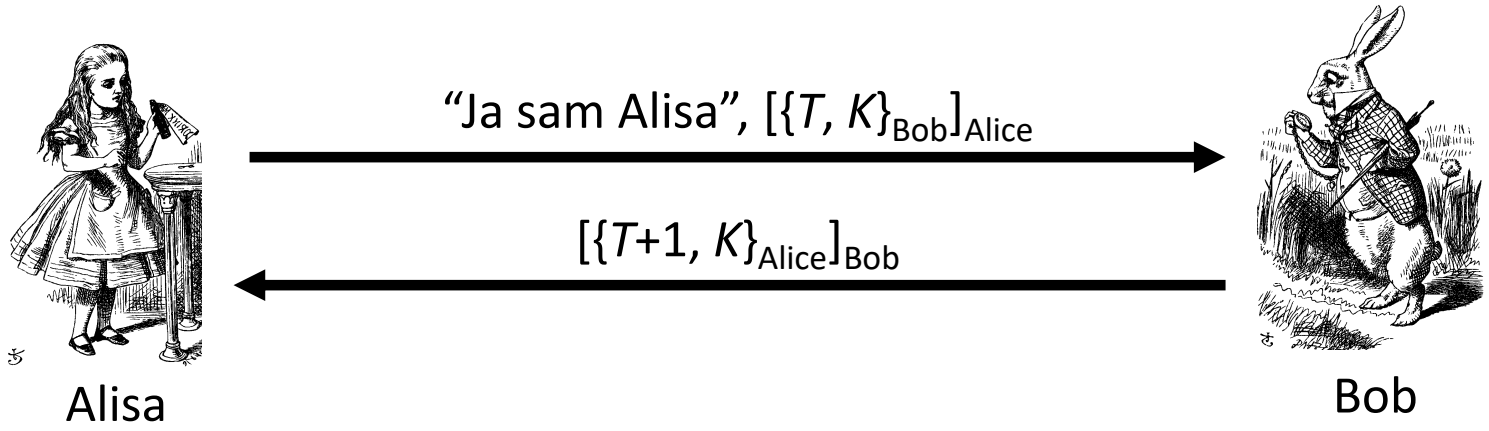
- Timestamp *T* predstavlja podatak o trenutnom vremenu.
- Koristi se u mnogim sigurnosnim protokolima (npr. Kerberos).
- Može da smanji broj poruka u autentifikaciji.
- Međutim, upotreba *T* vodi ka tome da je podatak o vremenu kritičan parametar po pitanju sigurnosti.
- Satovi gotovo nikad nisu sinhronisani, pa se mora se dozvoliti vremensko odstupanje
  - Iz ovoga proizilazi rizik ponovnog slanja iste poruke.
- Kako odrediti maksimalno vremensko odstupanje?

Autentifikacija javnim ključem i *timestamp T*.



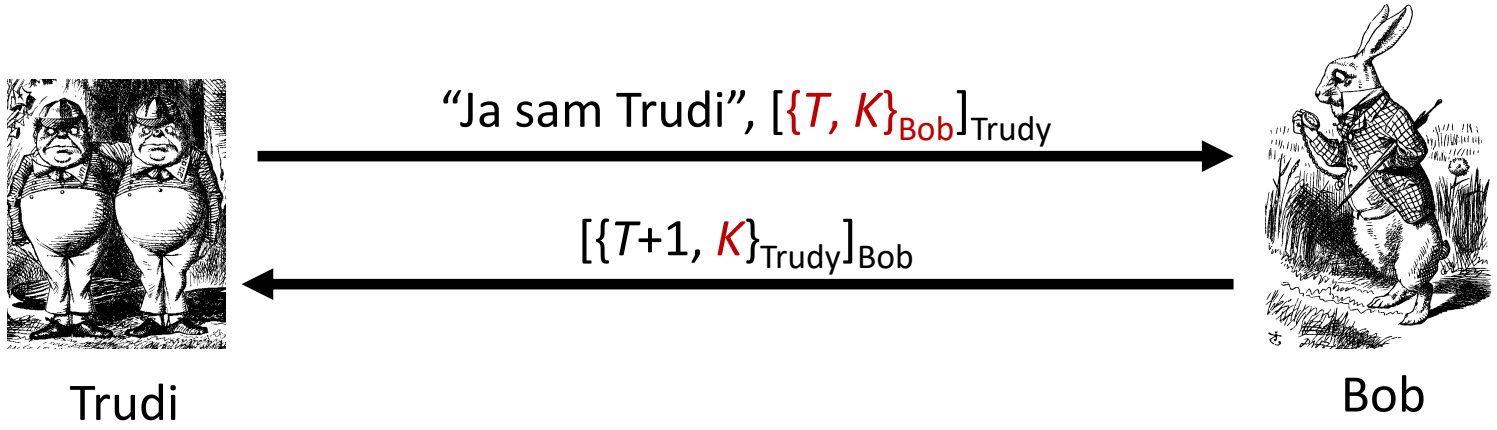
- Potpiši pa šifruj.
- Da li je ovaj protokol siguran?
- Izgleda da je sve u redu.

Autentifikacija javnim ključem i *timestamp T*.



- Šifruj pa potpiši.
- Da li je ovaj protokol siguran?
- Trudi zna Alisin javni ključ, tako da može da izračuna $\{T, K\}_{Bob}$, ...

Autentifikacija javnim ključem i *timestamp T*.



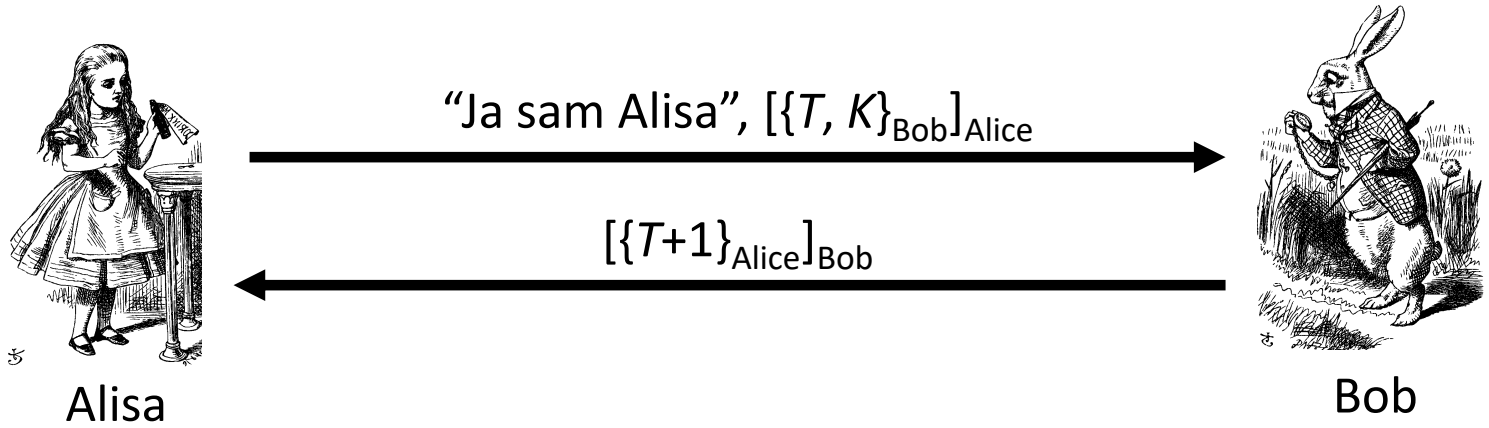
- Trudi na ovaj način dolazi do sesijskog ključa *K* koji koriste Alisa i Bob.
- Protokol "šifruj pa potpiši" nije siguran kada se koristi *timestamp T*.
- Napomena: u ovom slučaju Trudi mora da izvrši napad u okviru dozvoljenog vremenskog odstupanja.

Protokoli za autentifikaciju

Autentifikacija javnim ključem.

- Potpiši pa šifruj, upotreba slučajnih vrednosti → sigurno.
- Šifruj pa potpiši, upotreba slučajnih vrednosti → sigurno.
- Potpiši pa šifruj, upotreba *timestamp*-a → sigurno.
- Šifruj pa potpiši, upotreba *timestamp*-a → nije sigurno.

Rešenje problema šifruj pa potpiši uz upotrebu *timestamp*-a.



- Bob ne šalje ključ u odgovoru zato što ga Alisa već zna.
- Da li je ovaj protokol siguran?
- Izgleda da je sada sve u redu.

Autentifikacija i TCP

Autentifikacija zasnovana na TCP.

- TCP se u nekim slučajevima koristi za autentifikaciju.
- TCP nije namenjen da se koristi kao autentifikacioni protokol.
- Međutim, IP adrese u TCP konekciji se često koriste za autentifikaciju.
- Jedan mod IPSec koristi IP adrese za autentifikaciju.
  - To može da dovede do problema.

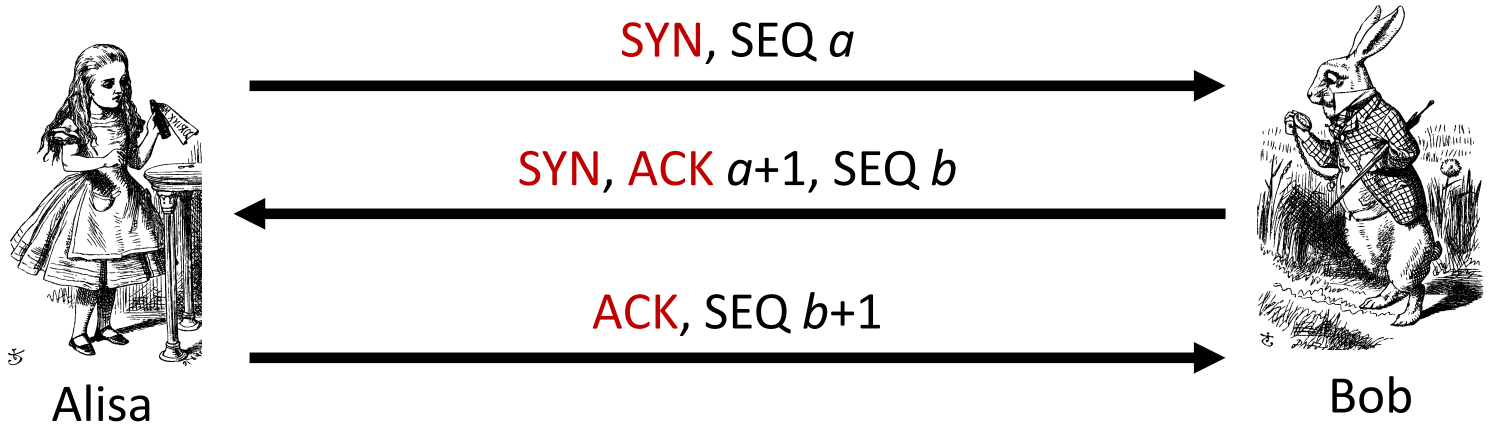
Autentifikacija i TCP

TCP *handshake*.

- Tri poruke:
  1. Alisa: Zahtev za sinhronizaciju SYN, SEQ *a*.
     - "*a*" treba da bude slučajna vrednost.
  2. Bob: Odobravanje zahteva SYN, ACK *a*+1, SEQ *b*
     - Bob zna Alisinu IP adresu.
     - "*b*" treba da bude slučajna vrednost.
  3. Alisa: Potvrda odobrenja ACK, SEQ *b*+1 (mogu se slati i podaci).
- Pretpostavka je da Trudi ne može da vidi drugi (SYN-ACK) paket.
- Nije baš jak oblik autentifikacije ali se koristi u praksi.

Autentifikacija i TCP

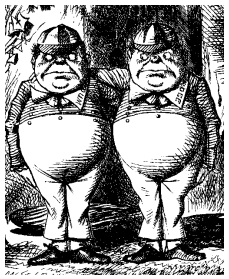
TCP *handshake*.



- Razmenjuju se tri poruke.
- Početna SEQ vrednost mora da bude slučajna.
- Zašto?

Napad na TCP autentifikaciju.

- Neka Trudi može da predvidi b_2 na osnovu b_1.



Trudi

1. SYN, SEQ = *t* (kao Trudi)
2. SYN, ACK = *t*+1, SEQ = b_1
3. SYN, SEQ = *t* (kao Alisa)

Bob

5. ACK = b_2+1, data

5.

5.

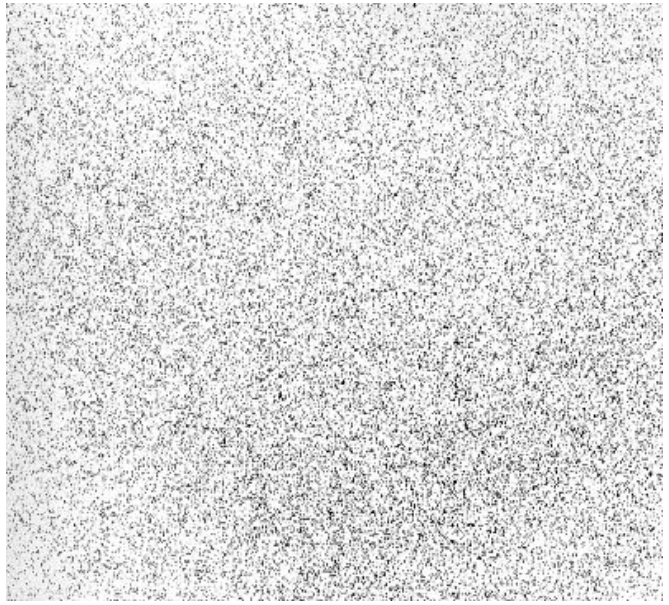
5.

5.

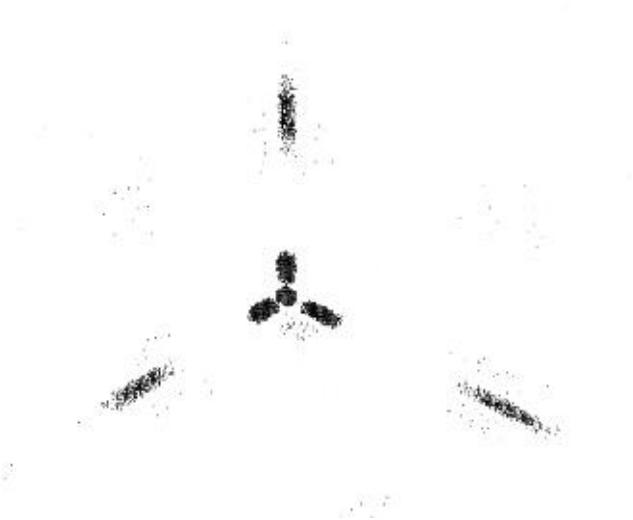
Alisa

4. SYN, ACK = *t*+1, SEQ = b_2

Napad na TCP autentifikaciju.



Slučajni SEQ brojevi



Inicijalni SEQ brojevi
na Mac OS X

- Ako inicijalne SEQ vrednosti nisu slučajne moguće je pogoditi inicijalni SEQ broj i prethodni napad je izvodljiv.

Autentifikacija i TCP

Napad na TCP autentifikaciju.

- Trudi ne može da vidi šta Bob šalje Alisi, ali može da šalje pakete serveru Bobu, predstavljajući se kao Alisa.
- Trudi mora da spreči Alisu da prima Bobove pakete (ili će se komunikacija prekinuti).
- Ako se koristi neka druga vrsta autentifikacije ovaj napad nema uspeha.
- Loše je zasnivati autentifikaciju na TCP.

Autentifikacija i TCP

Dobar protokol za autentifikaciju?

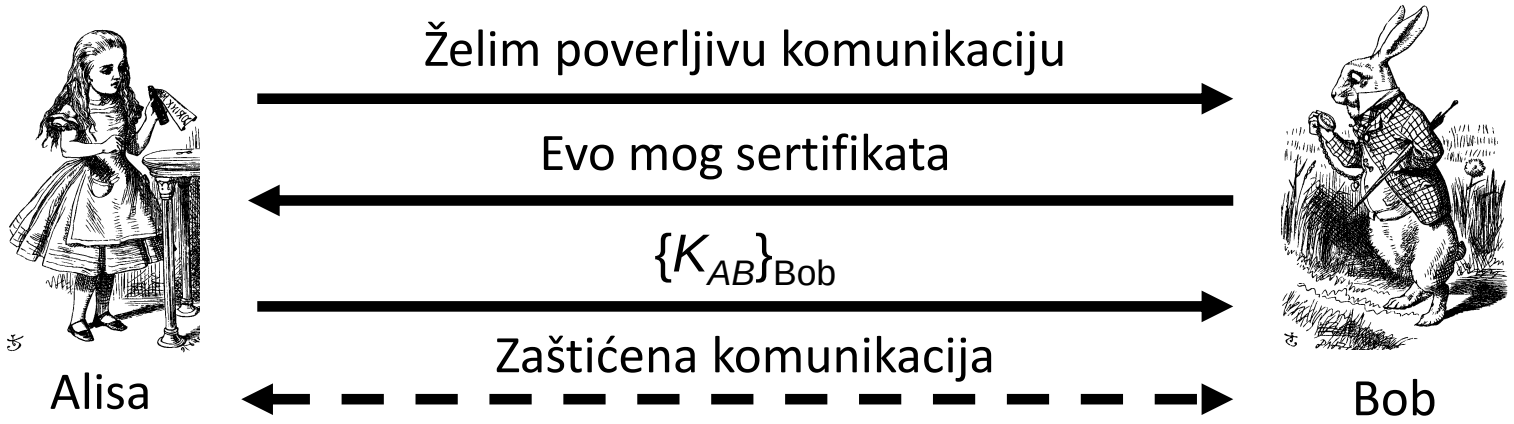
- Odluka zavisi od mnogo faktora:
  - Osetljivost primene.
  - Dozvoljeno kašnjenje.
  - Računska složenost.
  - Kriptografske funkcije koje se primenjuju (javni ključ, simetrični ključ, heš funkcije).
  - Da li se zahteva uzajmana autentifikacija?
  - Da li se zahteva primena sesijskog ključa?
  - ...

Sigurnosni protokoli u praksi

SSL.

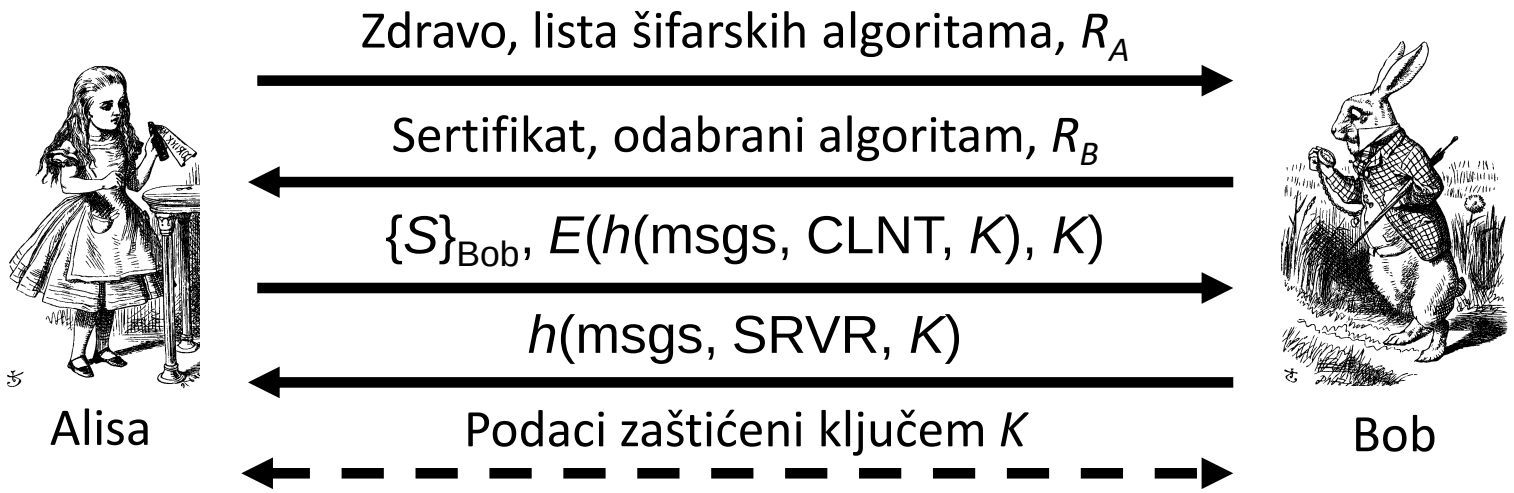
- SSL formira komunikacioni sloj koji se nalazi između aplikativnog i trasportnog sloja.
- SSL je protokol koji se koristi da obezbedi siguran prenos preko Interneta.
- Primer: želite da kupite knjigu sa www.amazon.com.
- Želite da budete sigurni da komunicirate baš sa Amazon-om (autentifikacija).
- Informacije o vašoj kreditnoj kartici moraju sa ostanu poverljive i neizmenjene u toku prenosa (poverljivost i integritet).
- Sve dok imate novac, Amazon verovatno neće zanimati ko ste vi (autentifikacija ne mora da bude uzajamna).

Jednostavan protokol.



- K_{AB} je simetrični ključ.
- Da li je Alisa sigurna da priča sa Bobom?
- Jedini način da to sazna je preko informacije da li Bob korektno dešifruje podatke.
- Da li je Bob siguran da priča sa Alisom?
- Ne, ali možda mu to nije ni potrebno.

Pojednostavljen SSL protokol.



- S je tajna vrednost, generiše je Alisa.
- $K = h(S, R_A, R_B)$, h – heš funkcija.
  - Bob nakon trećeg koraka može da izračuna ključ.
- msgs – sve prethodne poruke.
- CLNT i SRVR su konstante.
- Ako je četvrti korak uspešan, Alisa je autetnifikovala Boba.

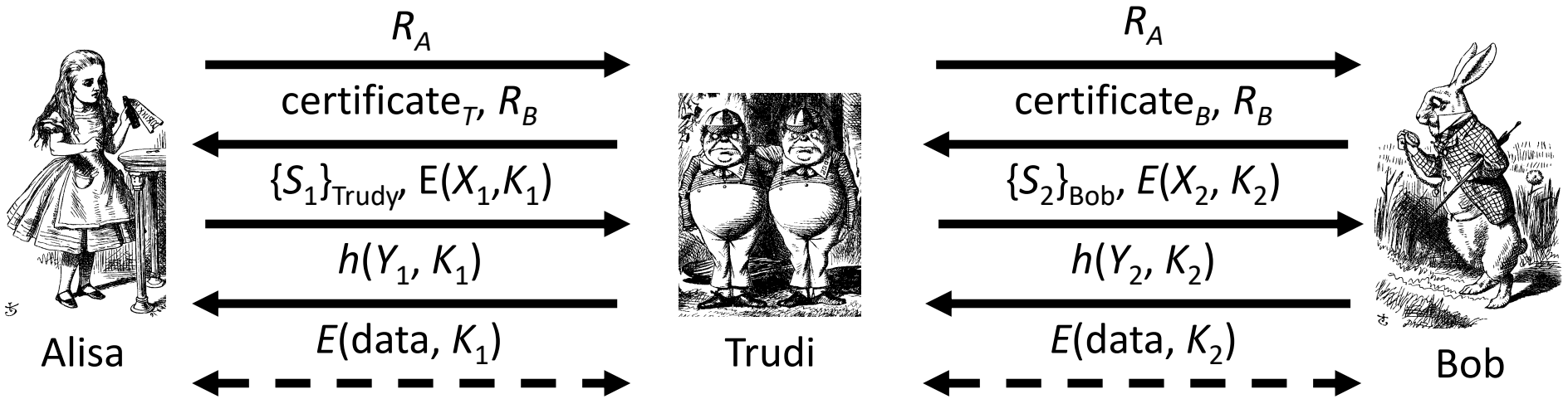
SSL ključevi.

- 6 “vrednosti” se dobija iz $K = h(S, R_A, R_B)$:
  - 2 ključa za šifrovanje – prijem i predaja,
  - 2 ključa za integritet (MAC) – prijem i predaja,
  - 2 inicijalne vrednosti IV – prijem i predaja,
- Zašto različiti ključevi u svakom pravcu?
  - Sprečavanje nekih napada.

SSL autentifikacija.

- Alisa je autentifikovala Boba, ali ne i obrnuto.
  - Kako klijent autentifikuje server?
  - Zašto server ne autentifikuje klijenta?
- Uzajamna autentifikacija je moguća: Bob šalje zahtev za sertifikatom u drugoj poruci.
  - Retko je potrebno.
  - Može se rešiti slanjem šifrovane lozinke (kada je ključ već razmenjen).

SSL i napad tipa čovek u sredini.

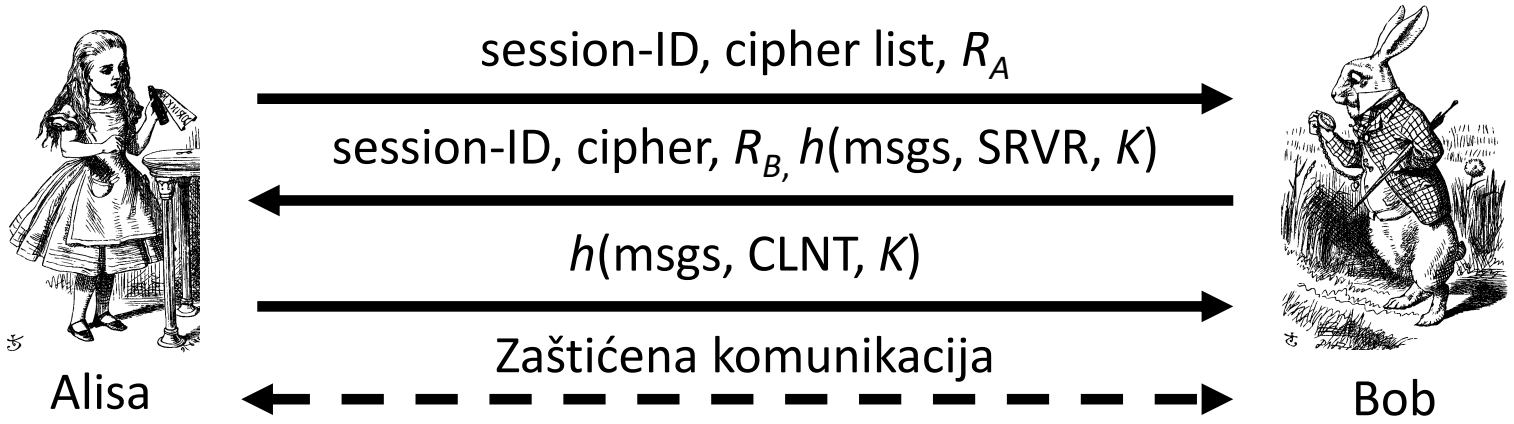


- Kako sprečiti ovaj napad?
- Bobov sertifikat mora biti potpisan od SA (npr. Verisign).
- Šta Web pretraživač radi ako potpis nije dobar? Šalje upozorenje (*warning*).
- Šta korisnik treba da uradi u tom slučaju? Raskid sesije!

SSL sesija i veza.

- SSL sesija je komunikacija koja se uspostavlja nakon razmene podataka koja je prikazana na prethodnim slajdovima.
- Uspostava sesija može da buda računaski zahtevna zbog primene kriptografije sa javnim ključem.
  - Ako se uspostavi SSL sesija za više HTTP konekcija, postoji problem.
- Omogućeno je da se u okviru jedne sesije uspostavi više veza (*connection*).
  - Sesija je opisana parametrima koji se dogovaraju u toku faze uspostave sesije između klijenta i servera.
  - Ti parametri su osnova za uspostavu svake nove veze.

SSL veza.



- Podrazumeva se da SSL sesija postoji.
- Vrednost tajne S je već poznata Alisi i Bobu.
- Ponovo je: $K = h(S, R_A, R_B)$.
- Nema operacija sa javnim ključem!

SSL – IPSec.

- IPSec:
  - Postoji na mrežnom nivou (deo je OS).
  - Nudi šifrovanje, integritet, autentifikaciju, ...
  - Prilično je složen.
- SSL:
  - Postoji na socket nivou (deo korisničkog prostora).
  - Nudi šifrovanje, integritet, autentifikaciju, ...
  - Jednostavni je.

IPSec.

- Dva dela IPSec:
  - IKE – *Internet Key Exchange.*
    - Uzajamna autentifikacija.
    - Razmena simetričnog ključa.
  - ESP/AH.
    - *Encapsulating Security Payload* – za šifrovanje i/ili integritet IP paketa.
    - *Authentication Header* – samo za integritet.

Kerberos.

- U Grčkoj mitologiji Kerber je troglavi pas koji čuva ulaz u Had.
- U sigurnosnom smislu, Kerberos je sistem za autentifikaciju zasnovan na simetričnom šifarskom sistemu.
  - Potekao je sa MIT.
  - Osnove su dali Needham i Schroeder.
  - Zasnovan je na poverenju u treću stranu (TTP).

Kerberos – motivacija.

- Autentifikacija koja koristi javne ključeve: *N* korisnika – *N* parova ključeva.
- Autentifikacija koja koristi simetrične ključeve: *N* korisnika zahteva oko N^2 ključeva.
- Kerberos koristi simetrične ključeve ali zahteva samo *N* ključeva za *N* korisnika.
  - Mora da postoji TTP.
  - Prednost: ne mora da postoji PKI.

Kerberos KDC.

- Centar za distribuciju ključeva (*Key Distribution Center*, KDC).
  - Predstavlja TTP.
  - Ne sme se dozvoliti da se TTP kompromituje!
- KDC deli simetrični ključ K_A sa Alisom, K_B sa Bobom, K_C sa Čarlijem, ...
  - Master ključ K_{KDC} je poznat samo KDC.
- KDC sprovodi autentifikaciju i generiše sesijske ključeve (ključeve za poverljivost i integritet).
- U praksi se mogu koristiti različiti simetrični algoritmi.

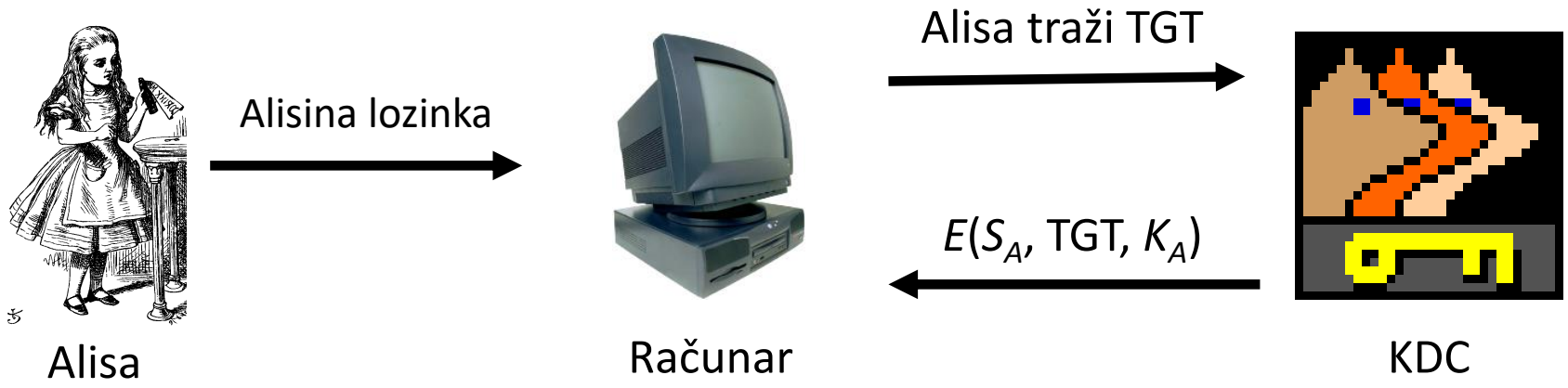
Kerberos tiket.

- KDC izdaje tiket koji sadrži informacije potrebne za pristup mrežnom servisu.
- KDC takođe izdaje *ticket-granting tickets* (TGTs) koji se koriste da bi se dobio tiket.
- Svaki TGT sadrži:
  - sesijski ključ,
  - korisnički ID,
  - vreme do kada je validan.
- Svaki TGT je šifrovan sa K_{KDC}.
- Jedino KDC može da pročita TGT.

Prijavljivanje na sistem.

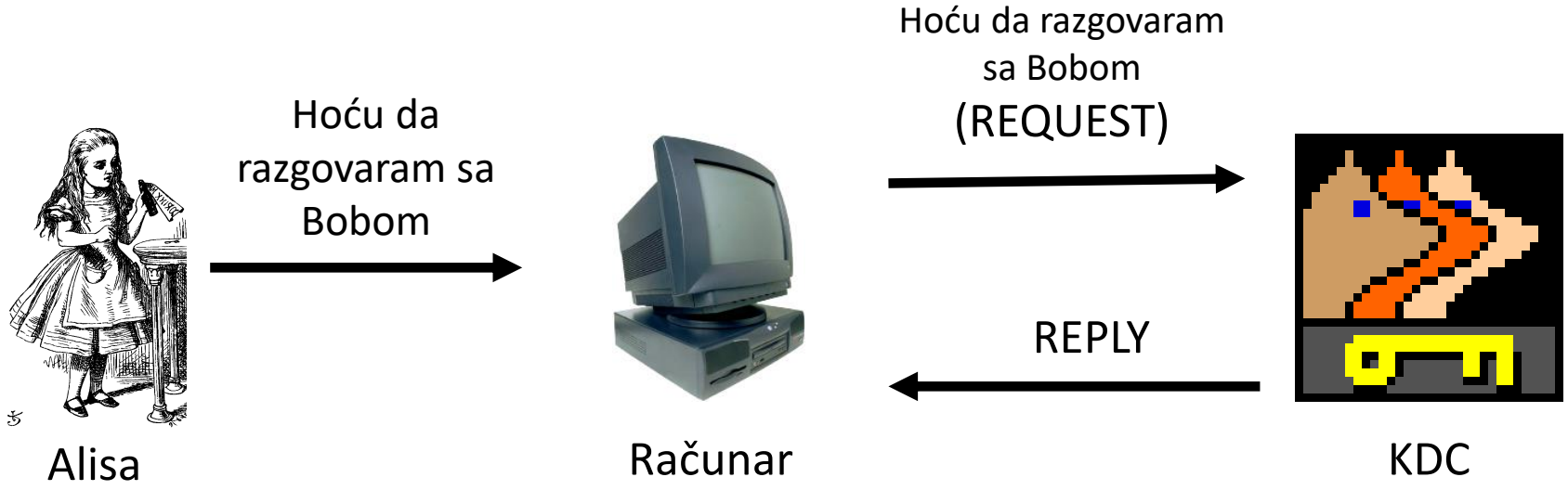
- Alisa unosi lozinku.
- Alisin računar:
  - izdvaja ključ K_A iz Alisine lozinke,
  - koristi K_A da bi dobio TGT za Alisu od KDC.
- Alisa, potom, može da koristi svoj TGT (potvrdu) za siguran pristup resursima na mreži.
- **Pozitivno**: Alisa nema mnogo posla oko sprovođenja procedure.
- **Negativno**: KDC mora biti od apsolutnog poverenja!

Kerberos – prijavljivanje na sistem.



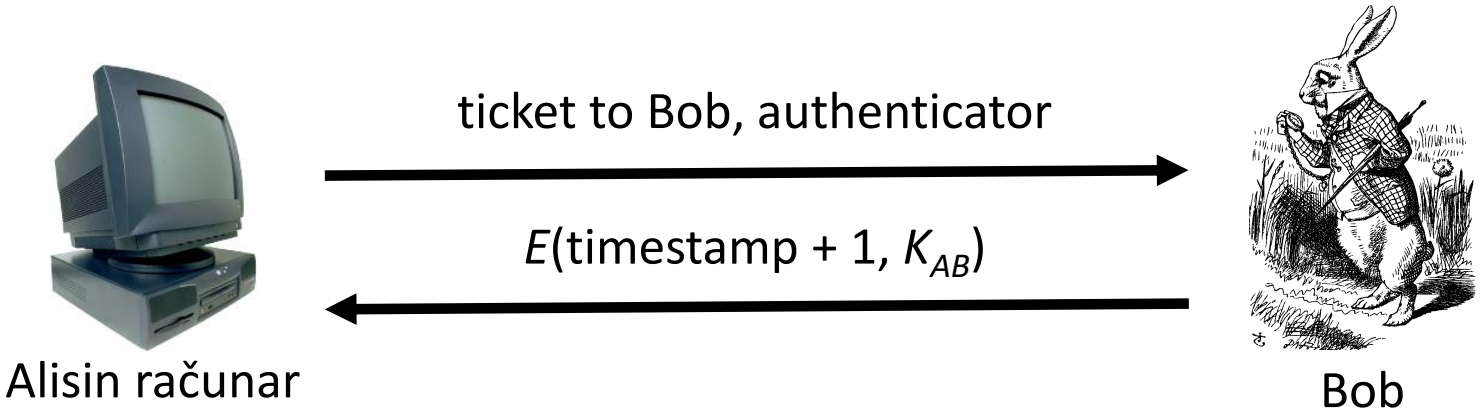
- Ključ K_A se dobija iz Alisine lozinke.
- KDC kreira sesijski ključ S_A.
- Računar dešifruje S_A, *TGT*, briše K_A.
- TGT = E(“Alice”, S_A, K_{KDC}).

Kerberos – Alisa traži tiket za Boba.



- REQUEST = (TGT, authenticator) gde je: authenticator = E(timestamp, S_A).
- REPLY = E("Bob", K_{AB}, ticket to Bob, S_A).
- ticket to Bob = E("Alice", K_{AB}, K_B)
- KDC uzima S_A iz TGT da bi verifikovao timestamp.

Kerberos – Alisa koristi tiket za Boba.



- ticket to Bob = E(“Alice”, K_{AB}, K_B).
- authenticator = E(timestamp, K_{AB}).
- Bob dešifruje “ticket to Bob” da bi dobio K_{AB} koji potom koristi za verifikaciju *timestamp*-a.

Literatura

1. M. Stamp (2006): *Information Security*. John Wiley and Sons.

Hvala na pažnji

Pitanja su dobrodošla.