# Kompiuterių tinklai - Autentifikacijos protokolai

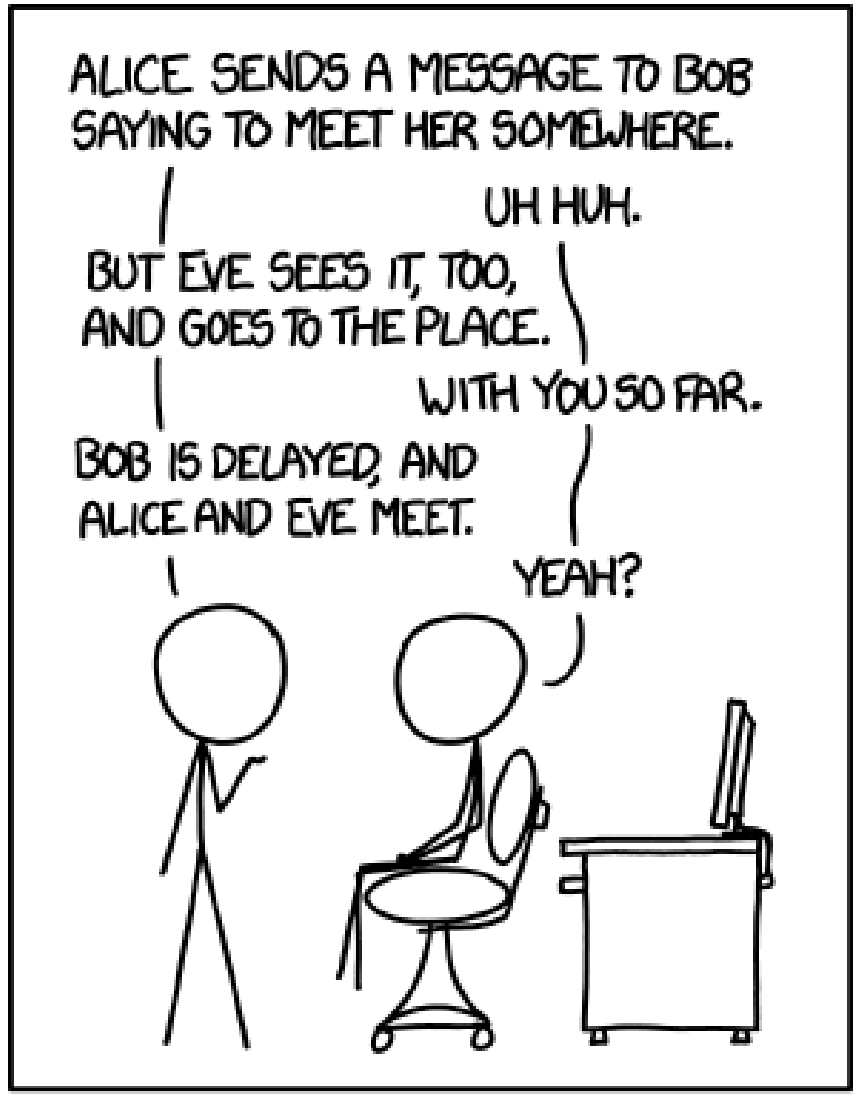
Vienuolikta paskaita (8 skyrius),
http://computernetworks5e.org/chap08.html

lekt. Vytautas Jančauskas

# Autentifikacijos protokolai

# Autentifikacija

- Autentifikacijos tikslas užtikrinti, kad šalis su kuria bendraujate yra ta kuri sako esanti.
- Nemaišykite autorizacijos ir autentifikacijos.
- Autorizacija - kas leidžiama konkrečiam procesui?
- Autentifikacija - ar tikrai bendraujame būtent su tuo procesu su kuriuo manome?
- Jeigu klientas Scott'o vardu liepia failų serveriui ištrinti failą "cookbook.old" iš serverio pusės reikia atsakyti į du klausimus:
  1. Ar tai iš tikro Scott'o procesas (autentifikacija)?
  2. Ar Scott'ui leidžiama ištrinti failą "coockbook.old" (autorizacija)?
- Tik atsakius į abu klausimus galima ištrinti failą. Tačiau iš principo svarbesnis pirmas klausimas. Kodėl?




I'VE DISCOVERED A WAY TO GET COMPUTER SCIENTISTS TO LISTEN TO ANY BORING STORY.

# Autentifikacijos protokolo eskizas

- Alice išsiunčia pranešimą arba Bob arba patikimam **KDC (Key Distribution Center)**.
- Alice pranešimas yra nuoširdus (jame pateikiama teisinga informacija).
- Toliau keičiamasi pranešimais abiejomis kryptimis.
- Trudy gali perimti tuos pranešimus, juos keisti, pakartoti ir t.t.
- Apsikeitimo pabaigoje turi būti užtikrinta, kad Alice būtų tikra, kad bendrauja su Bob, o Bob, kad jis bendrauja su Alice.
- Taip pat susitariama dėl slapto sesijos rakto, kuris naudojamas simetrinės kriptografijos algoritmuose (AES arba triples DES).
- Viešo rakto protokolai naudojami autentifikacijai ir tariantis dėl rakto.

# Žymėjimai

Visuose toliau nagrinėjamuose protokoluose naudosime tokius žymėjimus:

- $A, B$ yra Alice ir Bob identifikatoriai.
- $R_i$ yra iššūkiai, kur $i$ yra iššaukiančiojo identifikatorius.
- $K_i$ yra raktai, kur $i$ yra savininko identifikatorius.
- $K_S$ yra sesijos raktas.

# Autentifikacija naudojant bendrą slaptą raktą

- Tarsime, kad Alice ir Bob yra iš anksto sutarę dėl slapto rakto $K_{AB}$.
- Tokiu atveju protokolas atrodys taip:
  1. Alice siunčia savo identifikatorių A, Bob'ui. Bob'as savaime aišku nežino, ar identifikatorius atėjo iš Alice ar iš Trudy (žinutė 1).
  2. Bob'as parenka didelį atsitiktinį skaičių $R_B$ ir siunčia jį procesui prisistačiusiam Alice atviru tekstu (žinutė 2).
  3. Alice užšifruoja $R_B$ raktu $K_{AB}$ ir siunčia Bob'ui (žinutė 3).
  4. Bob'as atšifruoja gautą pranešimą ir tikrina ar atšifravus raktu $K_{AB}$ gaunasi jo išsiųstas skaičius. Jeigu gaunasi - vadinasi pranešimas atėjo iš Alice o ne iš Trudy, nes Trudy neturi rakto $K_{AB}$.
  5. Nors Bob žino, kad bendrauja su Alice, Alice nežino nieko. Jai atsakymus lygiai taip pat sėkmingai gali siųsti Trudy.
  6. Norint užtikrinti, kad Alice bendrauja su Bob, atliekamas tas pats procesas tik atvirkščia tvarka (žinutės 4 ir 5).

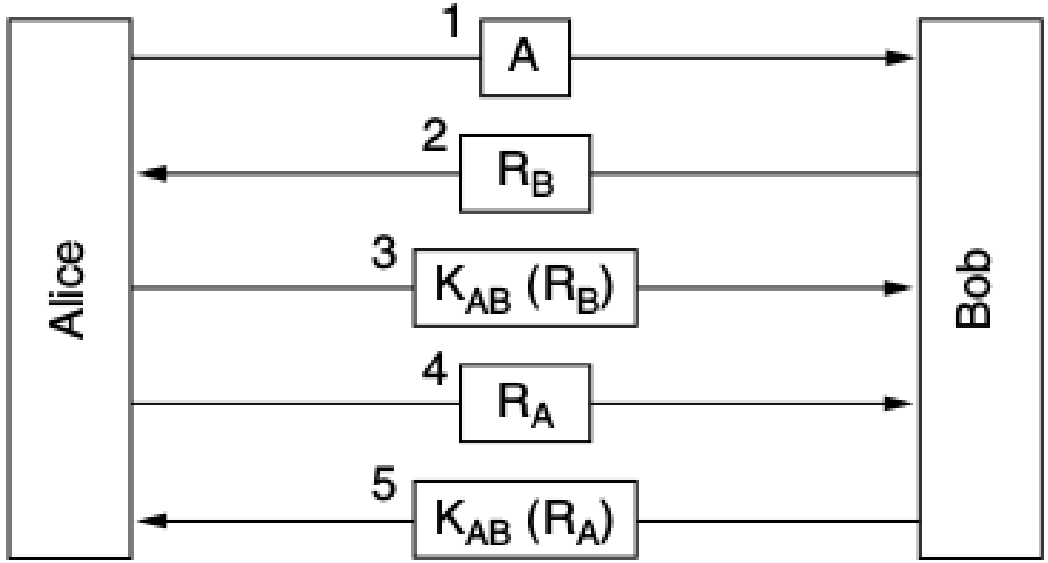



**Figure 8-32.** Two-way authentication using a challenge-response protocol.

# Sutrumpintas bendro slapto rakto protokolas

- Prieš tai aprašytam protokolui įgyvendinti reikia apsikeisti penkiomis žinutėmis. Ar galima protokolą sutrumpinti?
- Tarkime Alice kartu su savo identifikatoriumi nusiunčia savo atsitiktinį skaičių $R_A$., tokiu atveju užtenka trijų pranešimų.
  1. Alice išiunčia Bob savo identifikatorių ir atsitiktinį skaičių $R_A$ (žinutė 1).
  2. Bob išsiunčia užšifruotą $R_A$ ir savo atsitiktinį skaičių $R_B$ (žinutė 2). Alice dabar žino, kad bendrauja su Bob.
  3. Alice išsiunčia Bob užšifruotą $R_B$ (žinutė 3). Dabar Bob žino, kad bendrauja su Alice.
- Ar šis protokolas saugus?

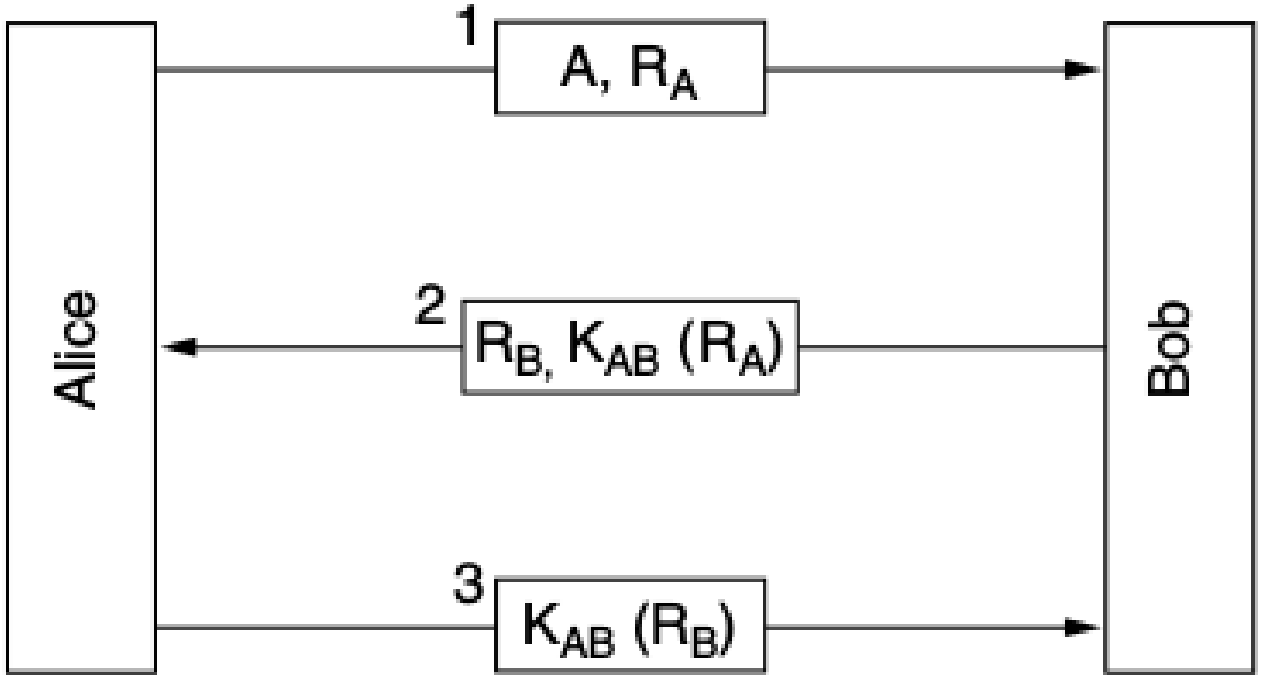



**Figure 8-33.** A shortened two-way authentication protocol.

# Atspindžio ataka (I)

- Atspindžio ataka galima, kai su Bob galima palaikyti vienu metu kelias sesijas (kas pilnai realu, jeigu Bob, pvz., bankas).
  1. Trudy apsimeta Alice ir siunčia Alice identifikatorių ir savo atsitiktinį skaičių $R_T$ Bob'ui (žinutė 1).
  2. Bob'as užšifruoja $R_T$ slaptu raktu $K_{AB}$ (nes galvoja, kad užmezginėja ryšį su Alice) ir siunčia Trudy kartu su savo atsitiktiniu skaičiumi $R_B$ (žinutė 2).
  3. Trudy pradeda kitą sesiją ir vėl apsimeta Alice ir siunčia Alice identifikatorių ir Bob'o atsitiktinį skaičių $R_B$ (žinutė 3).
  4. Bob'as atsako nauju atsitiktiniu skaičiumi $R_{B2}$ ir raktu $K_{AB}$ užšifruotu $R_B$ (žinutė 4).
  5. Trudy nutraukia antrą sesiją ir siunčia paskutinį pirmos sesijos atsakymą - atsitiktinį skaičių $R_B$ užšifruotą raktu $K_{AB}$ (žinutė 5). Dabar Bob, galvoja, kad bendrauja su Alice.
- Koks šios istorijos moralas?

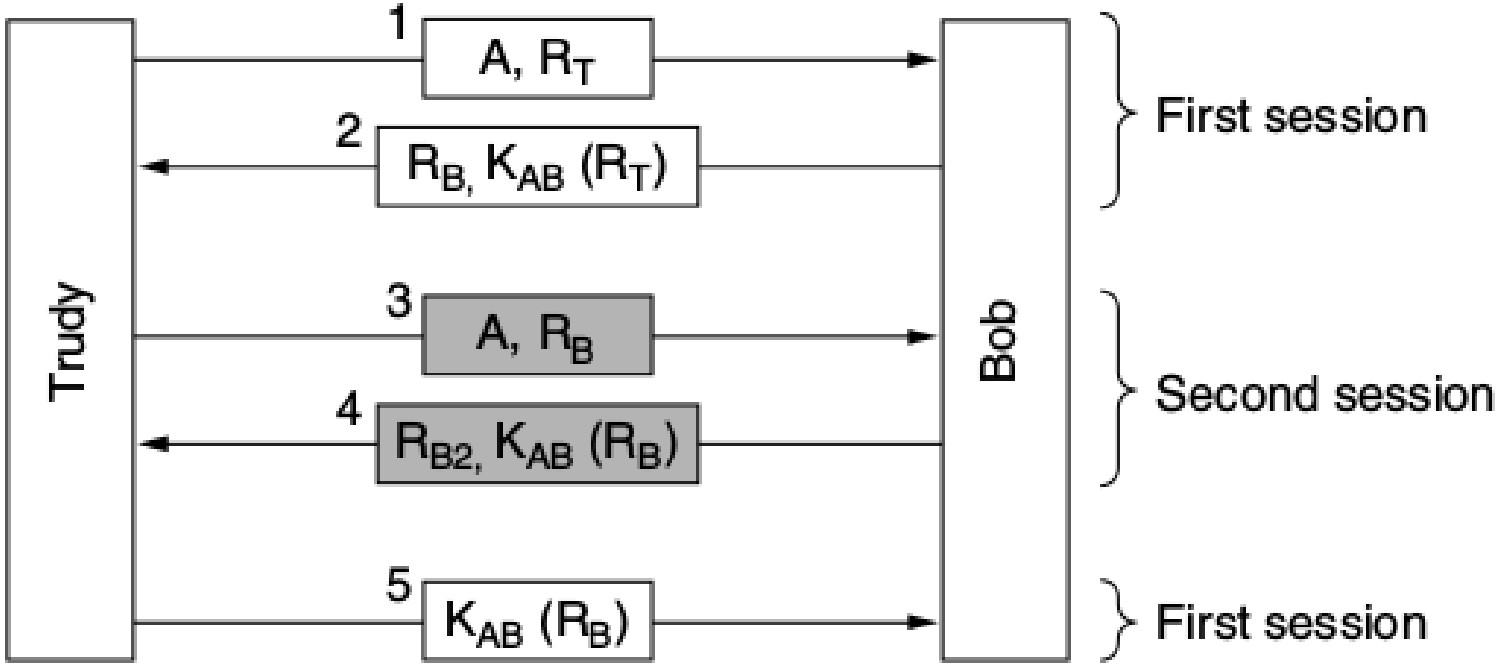



**Figure 8-34.** The reflection attack.

# Atspindžio ataka (II)

- Norint išvengti panašių protokolų kūrimo klaidų padeda šios keturios taisyklės.
  1. Sesijos iniciatorius turėtų įrodyti savo tapatybę prieš tai, kai ją įrodo atsakanti šalis. Taip Bob'as nepaviešins svarbios informacijos prieš tai kai Trudy turės įrodyti, kas ji yra.
  2. Iniciatorius ir atsakanti šalis turėtų naudoti du skirtingus raktus, net jeigu tai reiškia, kad reikia dviejų slaptų raktų $K_{AB}$ ir $K^*_{AB}$.
  3. Iniciatorius ir atsakanti šalis turėtų parinkti iššūkius iš skirtingų aibių (pvz. vienas naudoja lyginius o kitas nelyginius skaičius).
  4. Padaryti taip, kad informacija gauta vienos sesijos metu negalėtų būti naudojama kitos sesijos metu.
- Jei pažeidžiama bent viena iš šių taisyklių paprastai protokolas gali būti "nulaužtas".
- Kiek taisyklių pažeista mūsų nagrinėjamuose protokoluose?

# Atspindžio ataka (III)

- Ar galima atspindžio ataka kai naudojamas pirmas mūsų aptartas protokolas?
- Atspindžio ataka galima jeigu Alice taip pat leidžia kelias sesijas vienu metu (Alice yra ne žmogus o paslauga, pvz.). Tokiu atveju Trudy gali apsimesti Bob'u.
  1. Alice prisistato naudodama savo identifikatorių $A$ (žinutė 1). Alice nori užmegsti ryšį su Bob'u, tačiau jos žinutes perima Trudy.
  2. Trudy sukuria naują sesiją su Alice ir apsimeta Bob'u (žinutė 2).
  3. Alice atsako sukurdama atsitiktinį skaičių $R_A$ ir išsiųsdama Trudy (žinutė 3).
  4. Trudy tęsia pirmą sesiją ir išsiunčia Alice jos sugeneruotą skaičių $R_A$ atgal Alice kaip atsakymą į pirmą žinutę (žinutė 4).
  5. Alice užšifruoja šią savo pačios skaičių $R_A$ raktu $K_{AB}$ ir išsiunčia Trudy kaip pirmos sesijos trečią žinutę (žinutė 5).

# Atspindžio ataka (IV)

- Tęsiame ataką toliau:
  6. Trudy persiunčia Alice raktu $K_{AB}$ užšifruotą skaičių $R_A$ kaip antros sesijos trečią žinutę (žinutė 6). Praktiškai Trudy jau atliko visą darbą ir apgavo Alice. Dabar Alice mano, kad Trudy yra Bob.
  7. Alice siunčia atsitiktinį skaičių $R_{A2}$ kaip pirmos sesijos ketvirtą žinutę (žinutė 7).
  8. Trudy persiunčia atgal skaičių $R_{A2}$ kaip antros sesijos ketvirtą žinutę (žinutė 8).
  9. Alice užšifruoja $R_{A2}$ raktu $K_{AB}$ ir persiunčia Trudy, kaip antros sesijos penktą žinutę (žinutė 9).
  10. Trudy persiunčia $R_{A2}$ užšifruotą raktu $K_{AB}$ atgal Alice, kaip pirmos sesijos penktą žinutę (žinutė 10).
- Trudy sukūrė dvi autentifikuotas sesijas su Alice. Alice galvoja, kad abi sesijos yra su Bob.

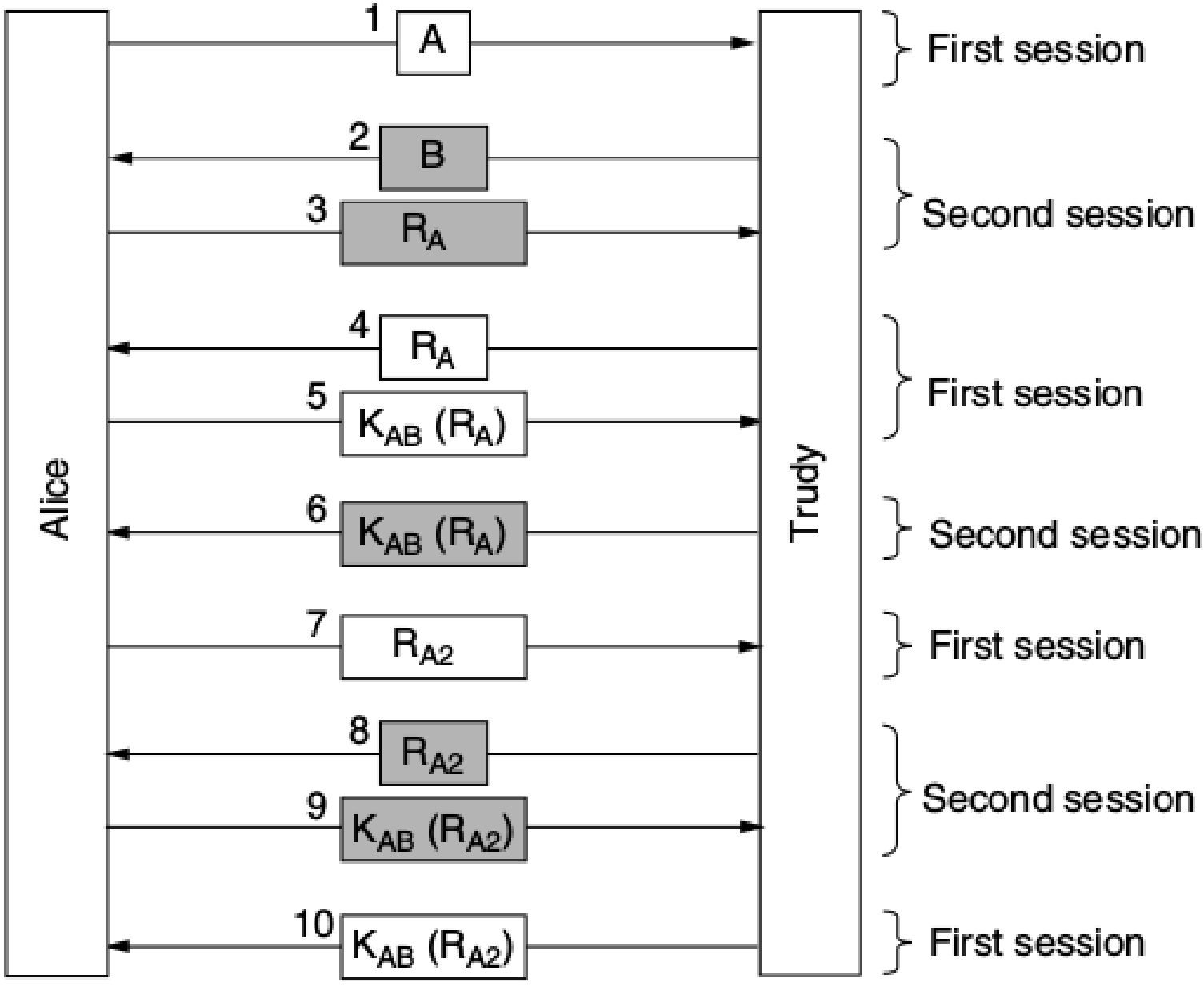



**Figure 8-35.** A reflection attack on the protocol of Fig. 8-32.

# HMAC protokolai

- Galimas paprastas autentifikacijos protokolas naudojantis **HMAC (Keyed-Hash Message Authentification Code)**.
  1. Alice sugeneruoja atsitiktinį skaičių (nonce) $R_A$ ir siunčia Bob (žinutė 1).
  2. Bob sugeneruoja atsitiktinį skaičių (nonce) $R_B$ ir siunčia Alice, kartu su **HMAC** sudarytu iš $R_A$, $R_B$, $A$, $B$ ir $K_{AB}$.
  3. Alice siunčia Bob'ui **HMAC** sudarytą iš $R_A$, $R_B$ ir $K_{AB}$.
- **HMAC** skaičiuojamas, paprasčiausiu atveju, sukonkatenuojant duomenis ir apskaičiuojant maišos funkcijos (pvz. SHA-1) reikšmę.
- Kodėl po šių veiksmų Alice tikra, kad bendrauja su Bob?
- Kodėl Bob tikras, kad bendrauja su Alice?
- Ar tenkinamos visos sąlygos kurias mes aptarėme?
- Ar įmanoma atspindžio ataka? Kodėl?

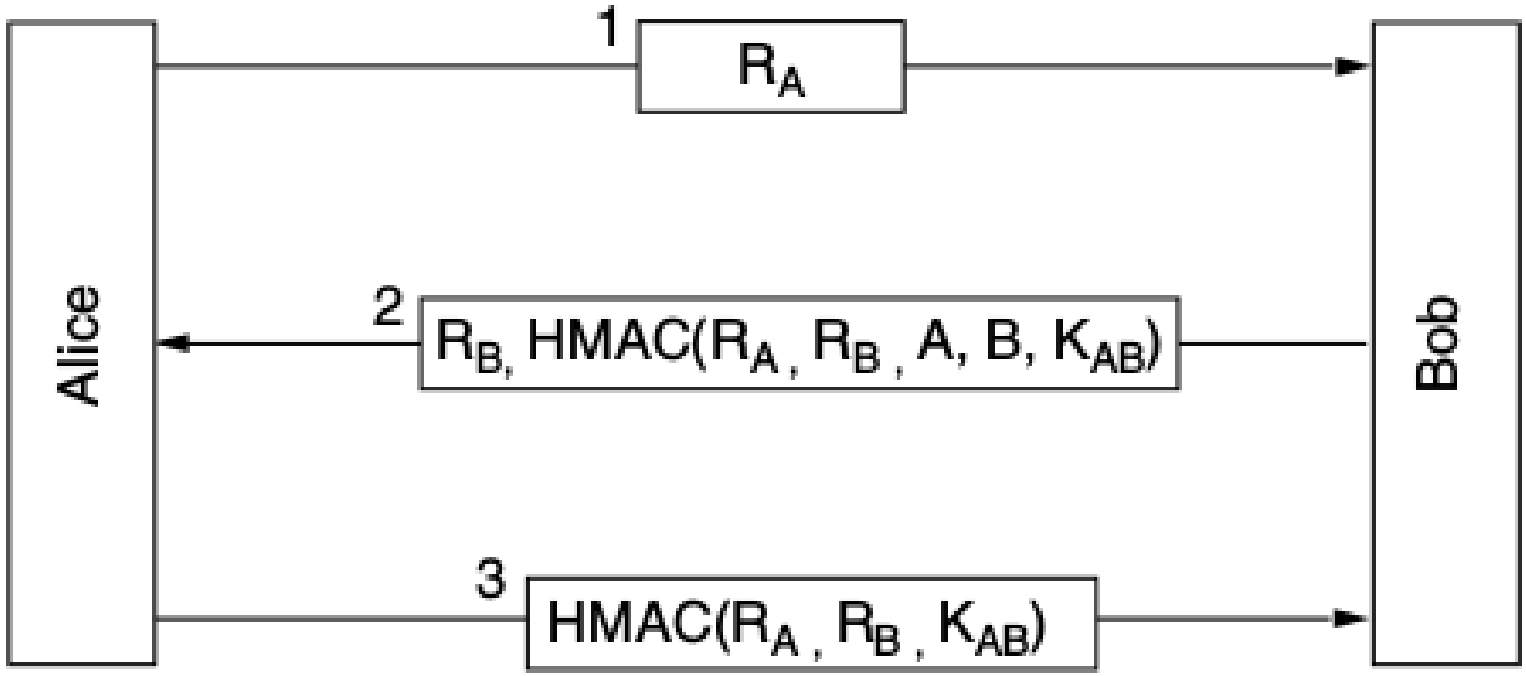



**Figure 8-36.** Authentication using HMACs.

# Apsikeitimo raktu protokolai (I)

- Prieš tai aptartuose protokoluose darėme prielaidą, kad Alice ir Bob susitarę dėl bendro slapto rakto.
- Tačiau kaip patogiai juo apsikeisti.
- Kaip bebūtų keista, apsikeisti slaptu raktu net ir naudojant nesaugų kanalą yra stebėtinai paprasta.
- Diffie-Hellman apsikeitimo raktais protokolas:
  1. Alice ir Bob susitaria dėl dviejų didelių skaičių $n$ ir $g$, kur $n$ yra pirminis, $\frac{n-1}{2}$ irgi yra priminis bei yra papildomų sąlygų $g$. Šie skaičiai gali būti vieši.
  2. Alice parenka didelį skaičių $x$ ir niekam apie jį nepraneša. Tą patį padaro Bob - jo skaičius yra $y$.
  3. Alice siunčia Bob'ui $(n, g, g^x \mod n)$.
  4. Bob atsako nusiųsdamas $g^y \mod n$.

# Apsikeitimo raktu protokolai (II)

- Alice iš Bob'o gautą skaičių pakelia laipsniu $x$ moduliu $n$ ir gauna $g^{xy} \mod n$.
- Bob'as iš Alice gautą skaičių pakelia laipsniu $y$ moduliu $n$ ir gauna $g^{xy} \mod n$.
- Taigi, jų bendras slaptas raktas yra $g^{xy} \mod n$.
- Trudy matė visas siunčiamas žinutes. Ar ji gali atkurti slaptą raktą?
- Trudy nelaimei, turint $g^{x} \mod n$ ir žinant $g$, jeigu skaičiai pakankamai dideli, neįmanoma atkurti $x$. Tas pats galioja ir $g^{y} \mod n$ atveju.
- Tarkime, $n = 47, g = 3, x = 8, y = 10$. Kaip atrodys perduodamos žinutės ir galutinis raktas?

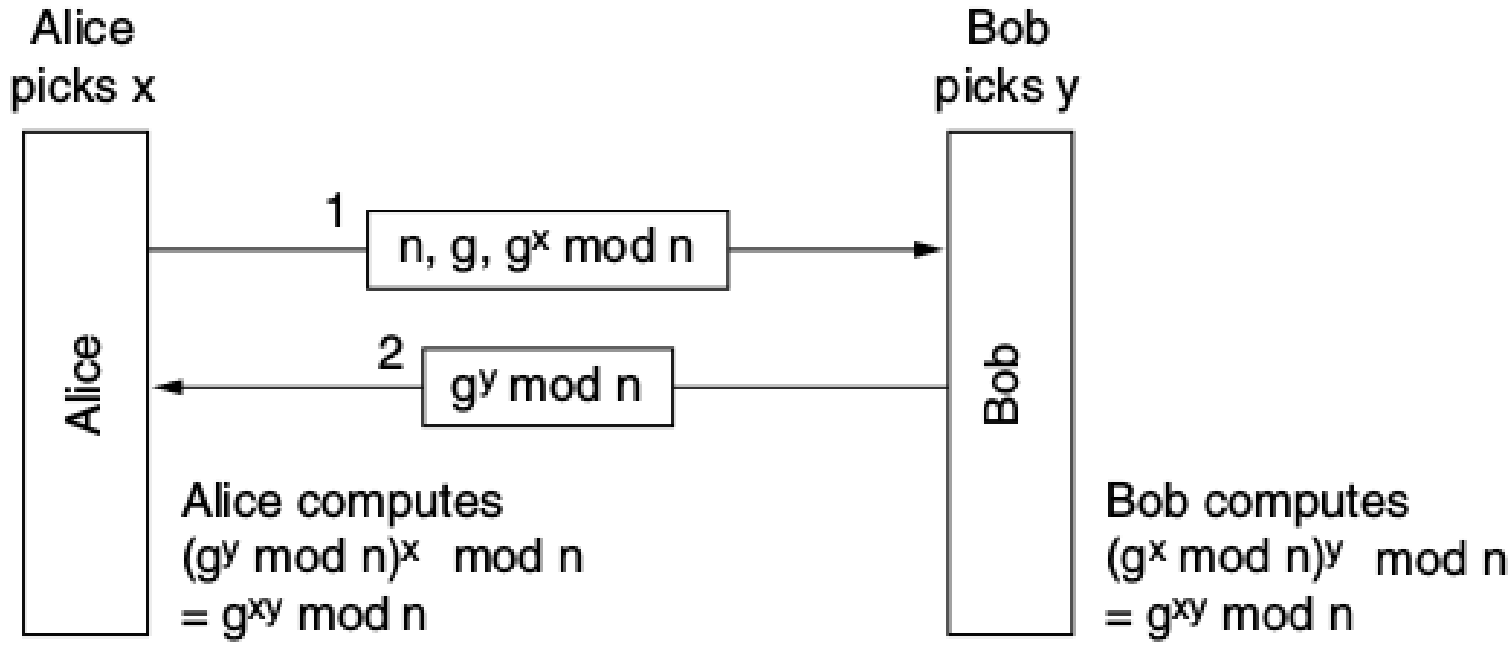



**Figure 8-37.** The Diffie-Hellman key exchange.

# “Man in the middle” ataka

- Tarkime Trudy įsiterpia tarp Alice ir Bob ir perima visus pranešimus.
- Trudy pasirenka savo atsitiktinį skaičių $z$, kol Alice ir Bob renkasi savo skaičius $x$ ir $y$.
- Alice siunčia žinutė nr. 1 Bob’ui, ją perima Trudy. Ji tada persiunčia Bob’ui žinutę su teisingais $g$ ir $n$, bet su savo $z$ vietoje $x$. Taip pat išsiunčia žinutę nr. 3 Alice, vėl gi naudodama savo $z$.
- Vėliau Bob susiųs Trudy žinutę nr. 4, kurią ji pasiliks sau.
- Dabar visi atlieka aritmetiką moduliu $n$. Alice gauna raktą $g^{xz}$ mod $n$, Bob’as gauna raktą $g^{yz}$ mod $n$. Trudy žino abu šiuos raktus, nes turi visus reikiamus komponentus.
- Dabar Trudy gali atkoduoti ir užkoduoti visą informaciją, kurią perduoda Alice Bob’ui ir atvirkščiai. Tiek Alice, tiek Bob atrodo, kad jie naudojasi saugiu kanalu.

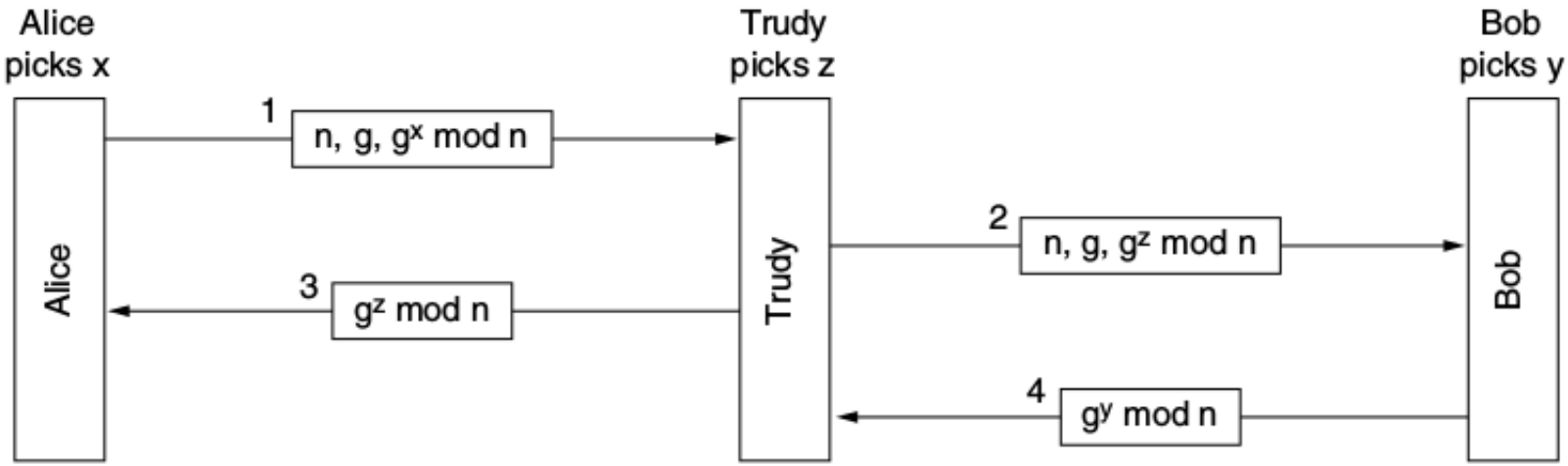



**Figure 8-38.** The man-in-the-middle attack.

# Raktų paskirstymas per raktų centrą (I)

- Kitas keitimosi raktais būdas yra naudoti raktų paskirstymo centrą.
- Šiuo atveju, kiekvienas naudotojas turi vieną raktą kuris yra bendras su KDC (Key Distribution Center).
- Paprasčiausias protokolas:
  1. Alice parenka sesijos raktą $K_S$ ir pasako KDC, kad nori bendrauti su Bob'u naudodama $K_S$. Ši žinutė užšifruojama slaptu raktu, kuris yra bendras tarp Alice ir KDC, $K_A$.
  2. KDC atšifruoja pranešimą, gauna Bob'o identifikatorių ir sesijos raktą.
  3. KDC sukonstruoja naują pranešimą, su Alice identifikatoriumi ir sesijos raktu.
  4. KDC užšifruoja šį pranešimą Bob'o raktu $K_B$ ir persiunčia Bob'ui.
- Autentifikaciją šiuo atveju gauname nemokamai. Kodėl?

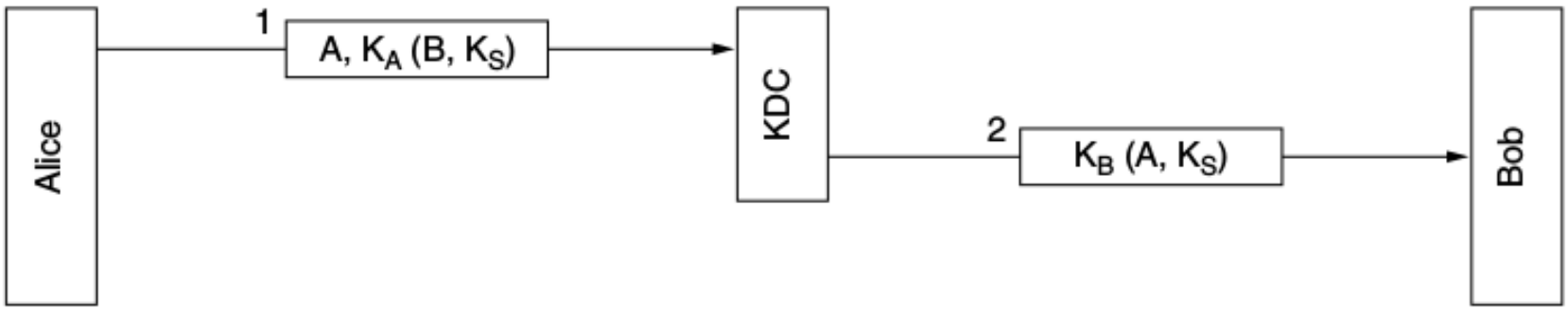



**Figure 8-39.** A first attempt at an authentication protocol using a KDC.

# Raktų paskirstymas per raktų centrą (II)

- Tarkime Trudy atlieka darbą Alice ir už tai Alice atsiskaito atliktama sujungimą su Bob'o banku ir nusiųsdama užklausą pervesti pinigus.
- Jeigu Trudy perima šias užklausas ir užklausą pervesti pinigus, Trudy gali kartoti šias užklausas, o Bob'as galvos kad užklausos iš Alice.
- Tokio tipo atakos vadinamos "replay" atakomis.
- Vienas variantas yra žymėti, kada atliekamos užklausos. Tada bandant atlikti jas iš naujo bus matoma, kad jos pasenusios. Tačiau išlieka laikrodžių sinchronizavimo problema.
- Kitas sprendimas yra įdėti nonce į pranešimus. Tada kievienas šalis prisimena nonce'us ir atmeta užklausas su pasikartojančiais.

# Needham-Schroeder autentifikacijos protokolas

- Alice praneša KDC, kad nori bendrauti su Bob, nusiųsdama atsitiktinį skaičių $R_A$ bei savo ir Bob'o identifikatorius $A$ ir $B$ (žinutė 1).
- KDC atsako atsiųsdamas skaičių $R_A$, Bob'p identifikatorių, sesijos raktą ir bilietą, kurį reikia persiųsti Bob'ui. Viskas užšifruojama Alice raktu $K_A$ (žinutė 2).
- Alice nusiunčia bilietą Bob'ui, kartu su nauju atsitiktiniu skaičiumi $R_{A2}$ užšifruotu sesijos raktu (žinutė 3).
- Bob'as užšifruoja $R_{A2} - 1$ sesijos raktu, prideda savo atsitiktinį skaičių $R_B$ ir persiunčia Alice (žinutė 4).
- Alice užšifruoja $R_B - 1$ sesijos raktu ir persiunčia Bob'ui (žinutė 5).
- Kodėl atimamas vienetas?

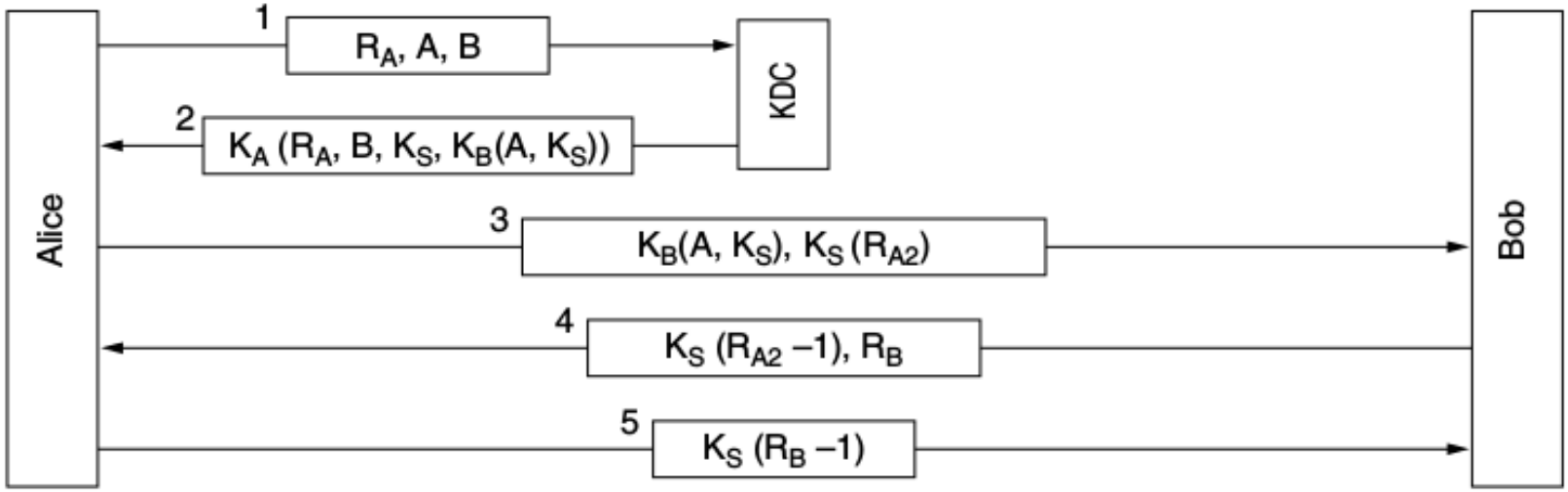



**Figure 8-40.** The Needham-Schroeder authentication protocol.

# Otway-Rees protokolas

1. Alice sugeneruoja porą atsitiktinių skaičių: $R$, kurį naudos kaip identifikatorių ir $R_A$ kurį naudos kaip iššūkį Bob'ui. Alice siunčia $A, B, R$ atviru tekstu ir tą pačią informaciją plius $R_A$ užšifruotus Alice raktu $K_A$ (žinutė 1).
2. Bob'as suformuoja analogišką žinutę su savo informaciją ir siunčia savo ir Alice informaciją KDC (žinutė 2).
3. KDC patikrina ar R yra vienodas abiejuose Bob'p pranešimo dalyse. Jei taip, sugeneruoja sesijos raktą, užšifruoja jį ir $R_B$ Bob'o raktu ir siunčia Bob'ui (žinutė 3).
4. KDC taip pat užšifruoja sesijos raktą ir $R_A$ Alice raktur ir siunčia Alice (žinutė 4).
5. Kaip Alice ir Bob žino, kad pranešimai yra iš KDC o ne iš Trudy?

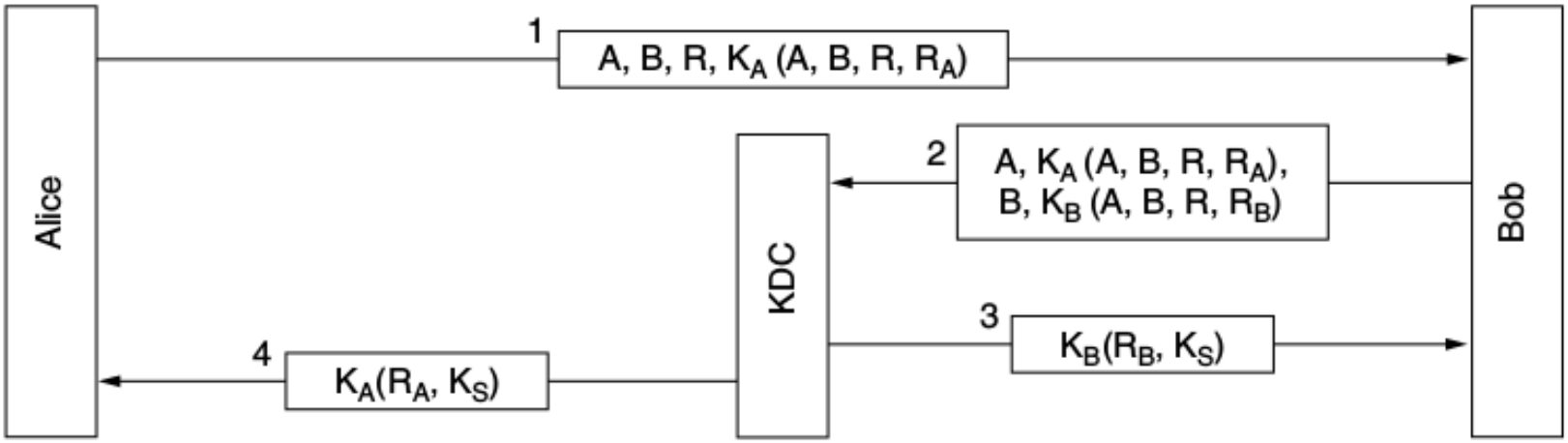



**Figure 8-41.** The Otway-Rees authentication protocol (slightly simplified).

# Autentifikacija naudojant Kerberos (I)

- Kerberos yra populiarus ir plačiai naudojamas Needham-Schroeder protokolo variantas.
- Pavadinimas kilęs iš daugiagalvio šuns saugančio Hado vartus graikų mitologijoje vardo.
- Kerberos buvo sukurtas MIT, kad naudotojai galėtų saugiai naudotis tinklo resursais.
- Kerberos veikimui užtikrinti reikia sinchronizuoti laikrodžius.
- Kerberos sistemą, be kliento (Alice) sudaro:
  1. Autentifikacijos serveris (AS): Tikrina naudotojus prisijungiant
  2. Bilietų išdavimo serveris (TGS): Išduoda identifikuojančius bilietus
  3. Bob'o: Teikia paslaugas, kurių reikia Alice.

# Autentifikacija naudojant Kerberos (II)

1. Alice prisėda prie terminalo ir suveda savo prisijungimo vardą $A$. Jos identifikatorius ir TGS vardas siunčiamas AS atviru tekstu (žinutė 1).
2. Atgal gaunamas sesijos raktas užšifruotas Alice raktu $K_A$ ir bilietas skirtas TGS, kuriame yra $A$, $K_S$ ir $t$ užšifruoti TGS raktu $K_{TGS}$ (žinutė 2).
3. Gavus žinutę nr. 2 jos paprašoma slaptažodžio. Slaptažodis naudojamas sugeneruoti $K_A$ kuriuo atšifruojamas gautas pranešimas ir gaunamas sesijos raktas.
4. Prisijungus Alice pasako, kad nori prisijungti prie Bob'o serverio. Siunčiamas pranešimas TGS, prašant bilieto bendravimui su Bob'u, taip pat TGS skirtas bilietas kurį Alice gavo iš AS (žinutė 3).
5. TGS sukuria sesijos raktą $K_{AB}$ ir persiunčia jį Alice. Siunčiamos dvi versijos, viena kurią gali atšifruoti tik Alice, kita, kurią gali atšifruoti tik Bob'as (žinutė 4).
6. Alice siunčia Bob'ui jo bilietą ir užšifruoja savo identifikatorių sesijos raktu $K_{AB}$ (žinutė 5).

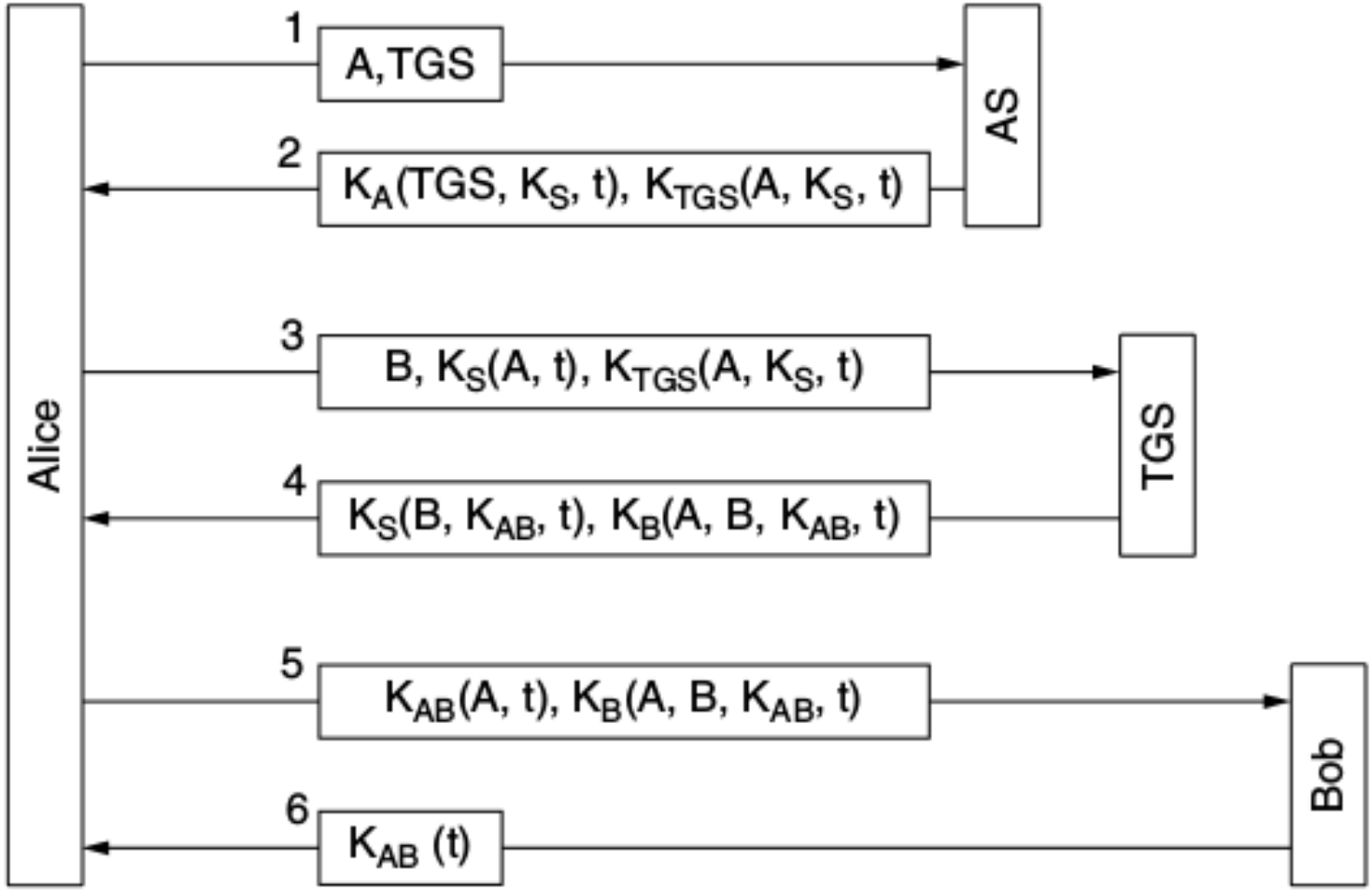



**Figure 8-42.** The operation of Kerberos V5.

# Autentifikacija naudojant viešą raktą

1. Užklausiama sertifikato su viešu Bob'o raktu iš direktorijos.
2. Direktorija gražina sertifikatą su pasirašytu Bob'o viešu raktu.
3. Alice patikrina ar parašas geras. Jei geras siunčia savo identifikatorių ir atsitiktinį skaičių $R_A$ Bob'ui, užšifruotą jo viešu raktu.
4. Bob'as užklausia Alice viešo rakto iš direktorijos.
5. Direktorija gražina viešą raktą.
6. Bob'as užšifruoja Alice skaičių $R_A$, savo skaičių $R_B$ ir sesijos raktą $K_S$ naudodamas viešą Alice raktą.
7. Alice gražina Bob'o skaičių $R_B$ užšifruotą sesijos raktu Bob'ui.

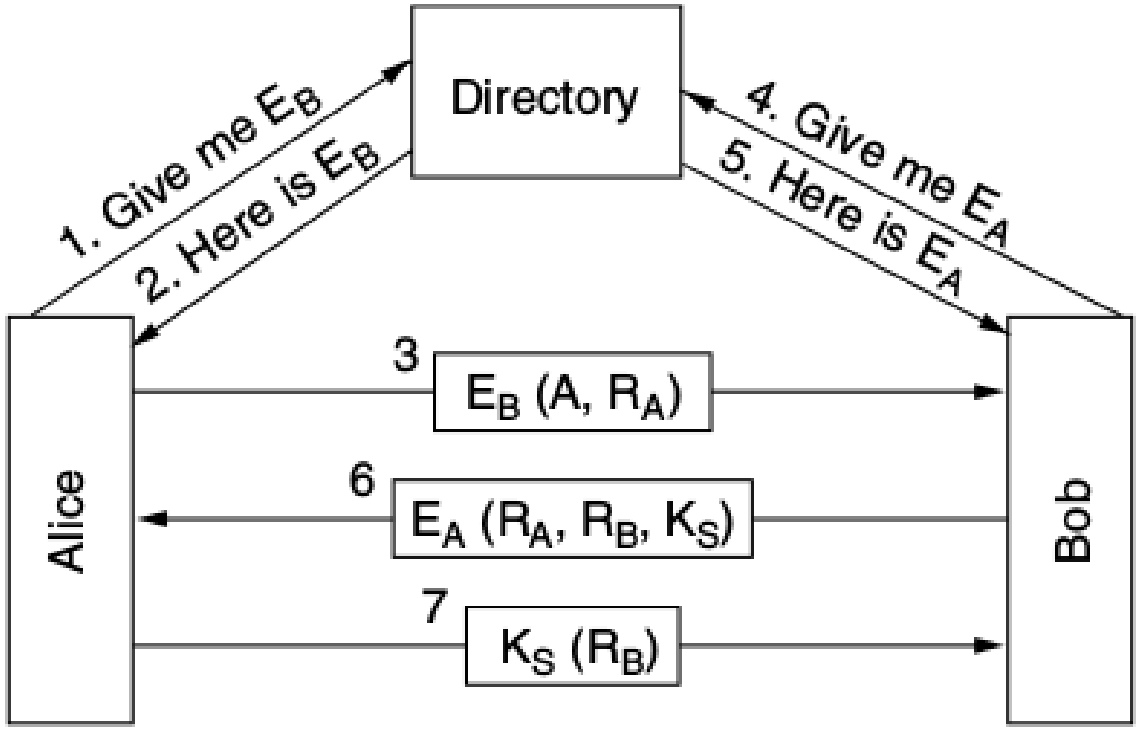



**Figure 8-43.** Mutual authentication using public-key cryptography.

# Užduotys

# Užduotys (I)

80. Kokius keturis reikalavimus turi tenkinti autentifikacijos protokolai?
81. Aprašykite kas yra atspindžio ataka.
82. Aprašykite kaip veikia autentifikacija naudojant HMAC.
83. Aprašykite kaip veikia Diffie-Hellman apsikeitimo raktais protokolas.
84. Aprašykite kaip atliekama “Man in the middle” ataka.
85. Aprašykite Needham-Schroeder autentifikacijos protokolą.
86. Aprašykite Otway-Rees protokolą.
87. Aprašykite kaip veikia Kerberos autentifikacija.
88. Aprašykite kaip veikia autentifikacija naudojant viešą raktą.

# Užduotys (II)

89. Pakeiskite vieną žinutę 12 skaidrių puslapyje esančiame protokole (8-34 pav.), kad jis taptų atsparus atspindžio atakai.
90. Alice ir Bob nori sukurti raktą Diffie-Hellman protokolu. Alice siunčia Bob'ui (227, 5, 82), Bob'as atsako (125). Alice slaptas skaičius $x$ yra 12, o Bob'o slaptas skaičius $y$ yra 3. Parodykite kaip sugeneruojamas slaptas raktas.
91. Naudojant Needham-Schroeder protokolą sugeneruojami du iššūkiai $R_A$ ir $R_{A2}$. Ar neužtenka vieno? Paaiškinkite.
92. Naudojant viešo rakto autentifikacijos protkolą septintoje žinutėje $R_B$ yra užšifruojamas naudojant $K_S$. Ar tai būtina ar užtektų siųsti kaip atvirą tekstą? Kodėl?