Identifikácia a autentizácia

Úvod do informačnej bezpečnosti (LS 2013/2014)

Michal Rjaško
rjasko@dcs.fmph.uniba.sk

Obsah

- Úvod
- Autentizácia na základe hesla
  - slabiny, prístupy, PIN, Passkey, jednorázové heslá
- Challenge-response autentizácia
  - zabezpečenie aktuálnosti, časové pečiatky
- Protokol na autentizáciu s využitím 3. strany
- Zero-knowledge autentizácia

Identifikácia a autentizácia

- Protokol medzi dvoma stranami
  - Dokazovateľ P
  - Overovateľ V
- Dokazovateľ P sa snaží dokázať svoju identitu overovateľovi V.
- Identifikácia: predstavenie identity P overovateľovi V
- Autentizácia: dôkaz / potvrdenie identity P
- Výstup protokolu:
  - akceptácia, t.j. identita P je pravá, komunikácia pokračuje
  - zamietnutie, ukončenie komunikácie
  - v niektorých prípadoch aj tzv. „session key“ – dočasný kľúč na šifrovanie danej relácie

Identifikácia a autentizácia - ciele

- **Korektnosť:**
  - V prípade poctivých strán P, V: V akceptuje identitu P
- **(Ne)prenositeľnosť:**
  - V nemôže zneužiť komunikáciu a vydávať za P pre tretiu stranu C
- **(Ne)falšovateľnosť:**
  - Žiadna tretia strana C sa nemôže vydávať za P pre V.
- **Robustnosť:**
  - Predchádzajúce vlastnosti zostávajú v platnosti aj v prípade veľkého množstva vykonaní protokolu.
- **“Real-time”:**
  - Autentizácia sa musí uskutočniť v realnom čase.

Autentizácia – základ

Autentizácia môže prebiehať na základe:

1. **Toho čo viem –** heslo, PIN, tajný kľúč
2. **Toho čo mám –** pas, kreditná karta, smart karta, token, mobil,…
3. **Toho čo som –** moje fyzikálne charakteristiky: odtlačok prsta, podpis, vzor dúhovky, hlas...

Využitie I & A

- Primárne využitie:
  - (Kontrolovaný) prístup k zdrojom
  - Logovanie / monitoring používateľov (kto čo robí)
  - Účtovanie (kto čo využíva)
- Ďalšie využitie:
  - Napr. vytvorenie session kľúča

Vlastnosti protokolov na I & A

- Reciprocita
  - jednostranná vs. vzájomná autentizácia
- Efektívnosť
  - Výpočtová náročnosť (# operácii)
  - Komunikačná zložitosť (# správ, prenesené bity)
- Zapojenie tretej strany
  - Dôveryhodná vs. nedôveryhodná 3. strana
  - Online vs. offline
- Bezpečnostné vlastnosti
  - Spôsob ukladania tajných hesiel / kľúčov
  - Dokázateľná bezpečnosť, Zero-knowledge

Fixné heslá

- Poskytujú tzv. slabú autentizáciu
- Zdieľané tajomstvo medzi používateľom a systémom
  - UserID používateľa identifikuje, heslo slúži ako dôkaz identity
  - Dôsledky:
    - Systém musí mať uložené heslá (v nejakej forme)
    - Používateľ musí systému ukázať svoje heslo (cez nejaký komunikačný kanál)

Schémy na I & A: Fixné heslá

Súbor s heslami v otvorenom tvare

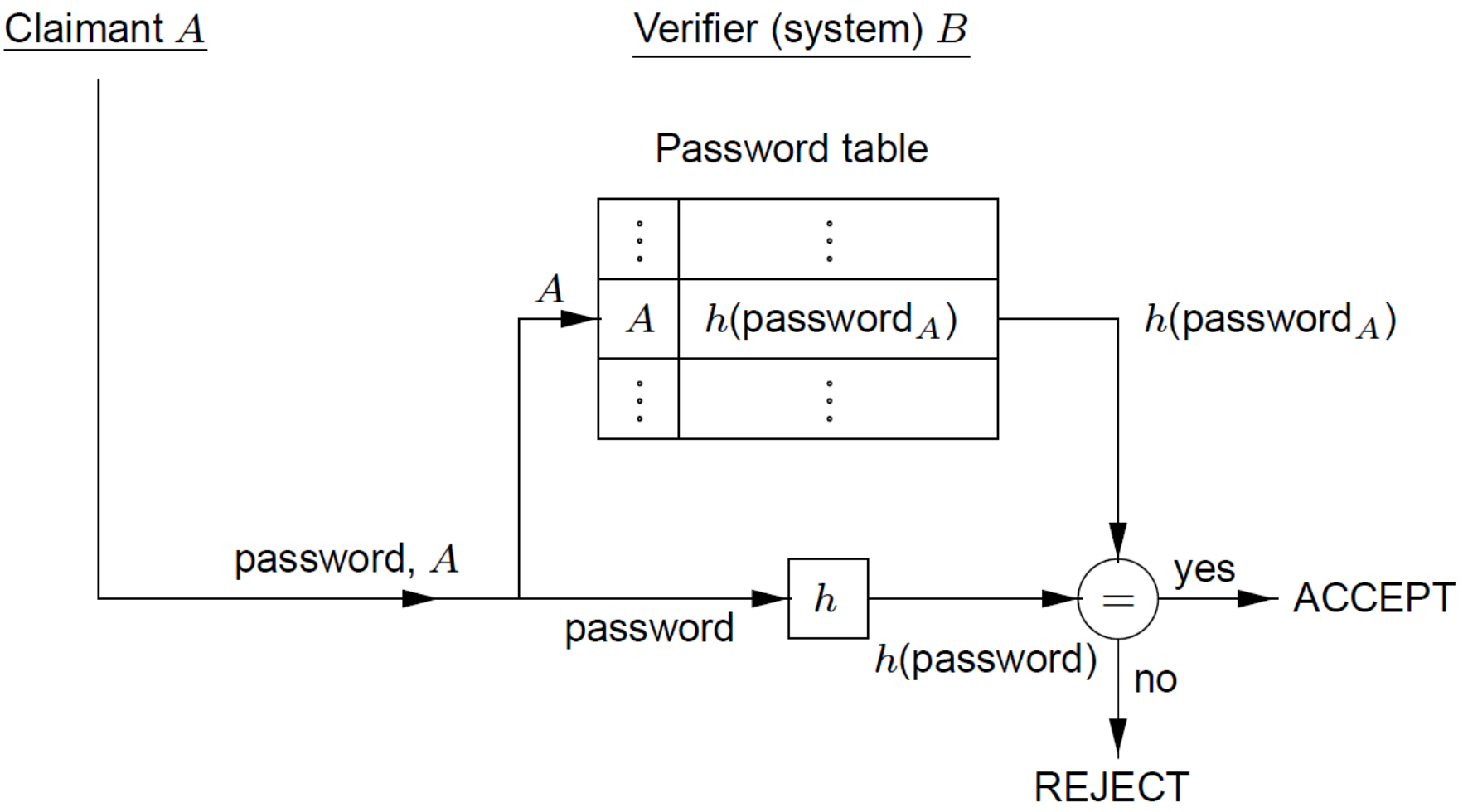
- Bez akejkoľvek ochrany súboru
  - Zjavne nebezpečné - ktokoľvek môže získať heslo
- Read / write ochrana v operačnom systéme
  - Používatelia posielajú heslá v otvorenom tvare
  - Administrátor / root ma prístup ku všetkým heslám
  - Backup súboru nemusí byť chránený
  - Heslo môže útočník odpočuť

Schémy na I & A: Fixné heslá

Zašifrované heslá

- Systém si pamätá (jednosmerný) haš hesiel
- Používateľ pošle heslo v otvorenom tvare
  - Systém ho zahašuje a porovná s uloženým záznamom
    - Administrátor už nevie získať žiadne heslo
    - Backup obsahuje iba haš hesiel
    - Heslo je však stále možné odpočuť
  - Alternatívne, používateľ heslo zahašuje a haš pošle na server
    - Útočník nevie odpočuť heslo
  - V obidvoch prípadoch však útočník môže zopakovať odpočutú správu

Overenie zahašovaného hesla



Fixné heslá: útoky

- Opakovaním
  - Ak je možné odpočúvať komunikáciu
- Úplné preberanie
  - Útočník skúša všetky možné heslá
  - Ochrana: zvýšiť veľkosť hesiel a / alebo limitovať počet (online) pokusov
  - Offline útok je stále možný:
    - útočník môže generované heslá porovnávať priamo so súborom (ak má prístup k súboru)

Fixné heslá: útoky

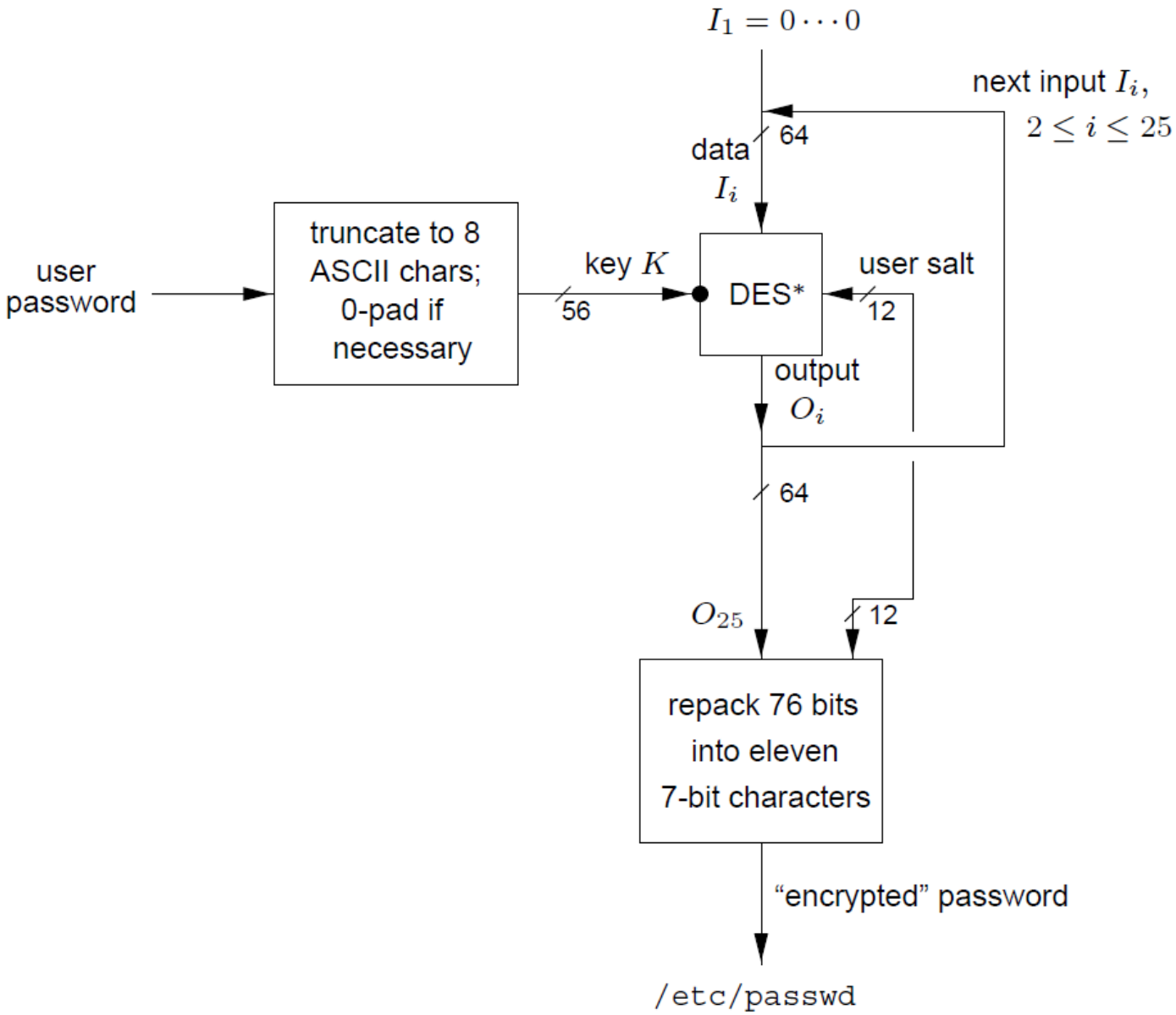
- Slovníkový útok
  - Väčšina používateľov si volí heslá z malej podmnožiny všetkých hesiel
  - Útočník skúša iba hesla zo slovníka – aj najväčší slovník má iba 250 000 slov, čo je menej ako 26^4.
  - Existujú aj špeciálne slovníky na „heslá“
  - Využiteľné najmä pri offline útokoch
    - V súčastnosti na to existujú šikovné programy – heslo odhalia v priebehu niekoľkých minút až hodín

Fixné heslá: útoky

Fixné heslá: ochrana voči útokom

- Kontrola sily hesla
  - Zabrániť používateľom zvoliť si slovníkové heslo.
- Použiť pomalú hašovaciu funkciu
  - Napr. iterovať štandardnú hašovaciu funkciu niekoľko krát
- Pridanie náhodnej soli
  - Pred zahašovaním hesla P k nemu pridáme náhodnú soľ S
  - C = h(S, P), zapamätáme si S, C
  - Dve rovnaké heslá majú rôznu soľ, t.j. rôzne šifrovanie
  - Zvačší sa zložiťosť slovníkového útoku (ale nie pre daného používateľa)
- Frázové heslá
- Expirácia hesiel

Heslá v UNIXe



Manažment hesiel

- Ako identifikovať používateľa ak ešte nemá heslo?
  - Ako ste dostali Vaše heslo pri nástupe na FMFI?
- Zabudnuté heslá
  - Zaslanie hesla nesprávnej osobne
  - Neposkytovať zabudnuté heslo volajúcemu, ale zavolajte naspäť na overené číslo používateľa
- Phishing
  - Dostanete email od banky vyžadujúci zmenu hesla

Manažment hesiel

- K obmedzeniam hesiel treba pristupovať rozumne
  - Ak heslo musí byť príliš zložité
    - používateľ si ho zapíše
  - Ak si heslo musí používateľ často meniť
    - zvolí si jednoduchšie heslo
  - Veľa systémov vyžadujúcich heslo
    - Koľko máte rôznych hesiel?
- Treba nájsť rovnováhu medzi bezpečnosťou a prívetivosťou pre používateľov

PIN

PIN – Personal Identification Number

- Podobné fixným heslám
- Používané spolu s nejakým tokenom, smart-kartou, kreditkou a pod.
- PIN je malý, zvyčajne 4-8 číslic
  - Môže a nemusí byť uložený v tokene (online vs. offline)
  - Môže byť odvoditeľný (hašovaním) z tajného kľúča a identity uloženej v tokene
  - Token obsahuje údaje na identifikáciu, PIN slúži na overenie vlastníctva tokenu – **dvojstupňová autentizácia**
- Na zamedzenie online útoku preberaním sa limituje počet nesprávnych pokusov.

Passkey

Password derived key

- Z PINu / hesla sa pomocou jednosmernej funkcie vygeneruje kľúč
- Kľúč je následne použítý na zabezpečenie komunikácie
- Overovateľ pozná PIN / heslo, môže si teda vygenerovať kľúč
- Možné skombinovať heslo so soľou – zakaždým nový kľúč
- Podobné slabiny ako v prípade fixných hesiel
  - Nutnosť pamätať si heslá na servery.

Jednorázové heslá

- Zdieľaný zoznam hesiel
  - Každý prvok použitý iba raz
  - Variácia: Tabuľka challenge-response dvojíc
    - Overovateľ pošle challenge, používateľ odpovie príslušným párom z tabuľky
- Sekvenčne aktualizované heslá
  - Začíname so zdieľaným heslom
  - Pri autentizácii s použitím hesla *i,* používateľ pošle nové heslo *i+1,* zašifrované heslom *i*

Jednorázové heslá: Lamportova schéma

- Sekvencie hesiel s využítím jednosmernej funkcie: **Lamportova schéma**
  - Set-up:
    - *P* má tajné heslo *w*. H je hašovacia funkcia
    - Určíme konštantu *t* – počet možných autentizácii
      - Po *t* autentizáciách je potrebné znovu vygenerovať *w*
    - *P* pošle *V* cez **autentický kanál** $w_0 = H^t(w)$
    - *V* inicializuje počítadlo pre *P,* napr. $I_P = 1$

Jednorázové heslá: Lamportova schéma

- i-ta iterácia Lamportovej schémy
  - P vypočíta $w_i = H^{t-i}(w)$ a pošle to *V*
  - V overí, či platí
    - $i = i_A$
    - $H(u) = w_{i-1}$, kde u je prijatá správa od *A*
  - Ak je overenie úspešné
    - V akceptuje heslo, zvýší i_P o 1
    - V uloží *u* ako w_i

Jednorázové heslá: Lamportova schéma

- Útok opakovaním nie je možný, avšak
  - Schéma je zraniteľná v prípade ak útočník získa *w* pred uskutočnením protokolu
    - Potrebujeme zabezpečiť autentický prenos H(w)
  - Problémy robia straty spojenia
- Výhoda
  - Malé komunikačné nároky
- Alternatívna schéma (vyžaduje si uloženie hesla na serveri)
  - A pošle serveru dvojicu *(r, H(r, P))*, kde *r* je zakaždým iné (napr. poradové číslo), *P* je zdieľané heslo

Challenge-response autentizácia

- Tzv. silná autentizácia
- Dokazovateľ dokáže znalosť nejakého tajomstva cez challenge-response protokol
  - Bez toho, aby tajomstvo počas protokolu odhalil (v niektorých prípadoch ho však overovateľ pozná)
- Dokazovateľ odpovedá na časovo závislý „challenge“
- Môže využívať
  - Symetrické šifrovanie
  - Asymetrické šifrovanie

Časovo závislé parametre

- Zamedzujú útokom opakovaním
- 3 základné typy:
  - Náhodné hodnoty
  - Sekvenčné číslovanie
  - Časové pečiatky
- “New and once” - nonce
  - Hodnota parametra musí byť zakaždým iná
  - Je potrebné zabezpečiť integritu parametrov – naviazať ich na ostatné posielané správy

Náhodné hodnoty

- Overovateľ V vygeneruje náhodnú hodnotu *r*
  - Pošle ju P ako „challenge“
- P odpovie správou, ktorá je „zviazaná“ s *r*
  - „zviazanosť“ s *r* zabezpečuje čerstvosť
- Problémy:
  - Opakovanie hodnoty *r* (narodeninový paradox)
  - Predvídateľnosť *r* – generovanie náhodných čísel nie je jednoduché
  - Komunikačná zložitosť
    - oproti časovým pečiatkam a sekvenčným číslam sa vyžaduje jedna správa naviac

Sekvenčné číslovanie

- Číslovanie správ vymenených medzi P a V
  - Monotónne rastúce číslovanie
- Problémy:
  - Potreba dlhodobo uchovávať aktuálne poradové číslo správy
  - Synchronizácia
  - Potreba riešiť výpadky spojenia a pod.
  - Nemožnosť detekovať „forced delay“ útok

Časové pečiatky

- Do každej posielanej správy zakomponujeme časovú pečiatku
  - Akceptujeme len správy s časovou pečiatkou, ktorá je v rámci nejakého akceptovateľného časového okna
  - Môžu sa využívať aj na časové obmedzenie prístupu
  - Umožňujú detekciu „forced delay“ útokov
- Nevýhody
  - Nutná synchronizácia hodín
    - Ak je synchronizácia vykonaná po sieti, je potrebné komunikáciu zabezpečiť – zase s využitím časových pečiatok?
  - Potreba ukladať prijaté časové pečiatky v rámci daného časového okna
    - Aby sme vedeli zabrániť útokom opakovaním

Challenge-response autentizácia

s využitím symetrického šifrovania

- Obe strany A,B zdieľajú nejaký tajný kľúč k
- Základné (jednoduché) protokoly ISO/IEC 9798-2:
  - S využitím časových pečiatok:
    - $A \rightarrow B : E_k(t_A, B)$
    - Po prijatí, B správu dešifruje a overí časovú pečiatku
    - Posielanie identifikácie druhej strany zamedzuje použitia rovnakej správy na autentizáciu B do A
  - S využitím náhodných čísel
    - $B \rightarrow A : r_B$
    - $A \rightarrow B : E_k(r_B, B)$
    - Po prijatí, B správu dešifruje a overí r_B (nemalo by sa opakovať).

Challenge-response autentizácia

s využitím symetrického šifrovania

- Vzájomná autentizácia
  - B → A : r_B
  - A → B : $E_k(r_A, r_B, B)$
    - Po prijatí B správu dešifruje a skontroluje r_B
  - B → A : $E_k(r_B, r_A)$
    - Po prijatí A správu dešifruje a skontroluje r_A
- S využitím hašovacích funkcií (ISO/IEC 9798-4):
  - B → A : r_B
  - A → B : $r_A, h_k(r_A, r_B, B)$
    - Po prijatí B zahašuje r_A, r_B, B a porovná s prijatou správou
  - B → A : $h_k(r_B, r_A)$
    - Po prijatí A zahašuje r_A, r_B a porovná s prijatou správou

Challenge-response autentizácia

s využitím **asymetrických** techník

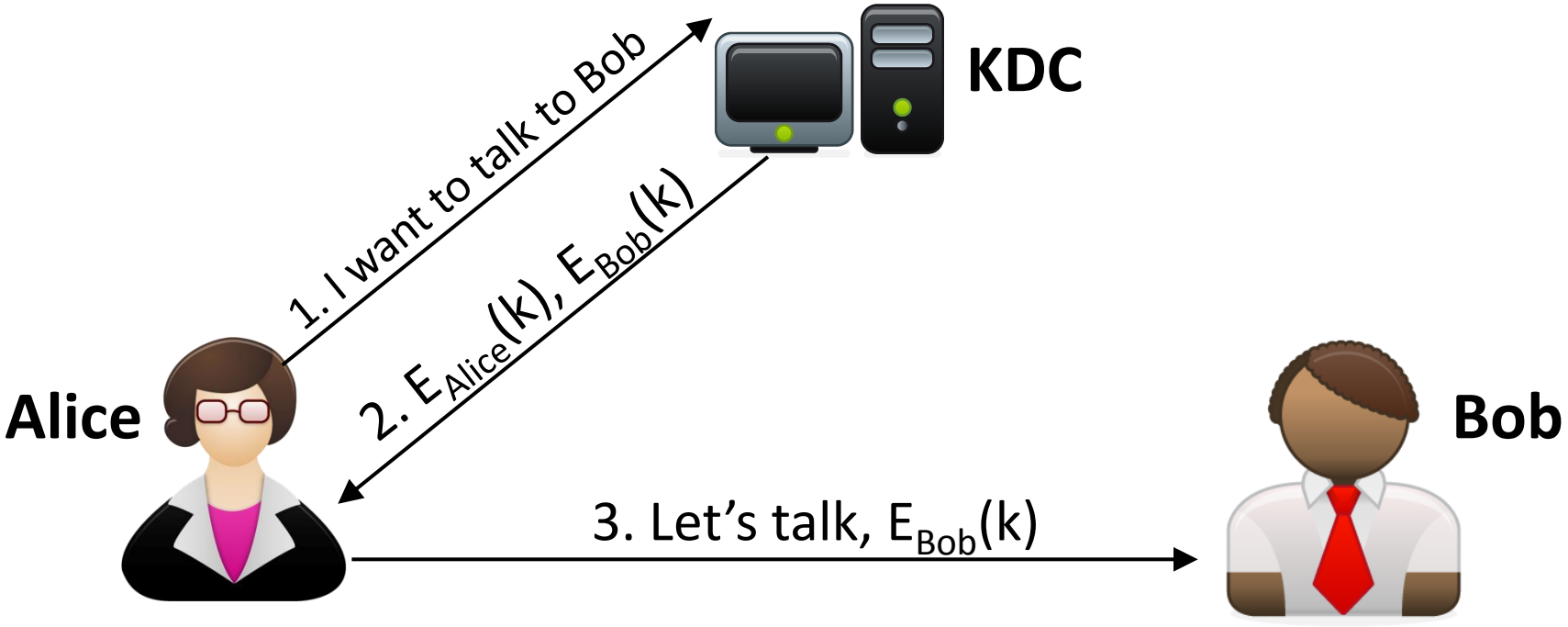
- Vzájomná autentizácia, asymetrické šifrovanie
  - B → A : $Pub_A(r_B, A)$
    - Po prijatí, A dešifruje a získa r_B
  - A → B : $Pub_B(r_A, r_B)$
    - Po prijatí, B dešifruje, získa r_A, r_B a porovná r_B
  - B → A : r_A

Základné challenge-response protokoly

- Všetky uvedené protokoly
  - Sú dvojstranné protokoly, t.j. bez tretej strany
  - Dokazovateľ aj overovateľ si dôverujú
  - Predpokladajú distribúciu kľúčov medzi komunikujúcimi stranami
    - T.j. strany sa navzájom poznajú, zdieľajú tajný kľúč / poznajú verejný kľúč druhej strany
    - Problematické v prípade veľkého množstva komunikujúcich párov
- Ak posledný predpoklad nie je platný
  - Je potrebné využiť 3. stranu na výmenu / distribúciu kľúčov – napr. KDC (Key Distribution Center)

Key distribution center

- Server na distribúciu kľúčov
- Každý používateľ zdieľa so serverom tajný kľúč



Needhamov-Schroederov protokol

- Autentizácia s využitím 3. strany
  1. A → S: A, B, r_A
  2. S → A: $E_{K_{AS}}$ (r_A, B, K_{AB}, $E_{K_{BS}}$(K_{AB}, A))
     - A dešifrjue. Overí aktuálnosť r_A. Získa kľúč K_{AB}
  3. A → B: $E_{K_{BS}}$ (K_{AB}, A)
     - B dešifruje, získa kľúč K_{AB}
  4. B → A: $E_{K_{AB}}$(r_B)
     - A dešifruje a získa r_B.
  5. A → B: $E_{K_{AB}}$(r_B-1)
     - B dešifruje, overí aktuálnosť cez r_B-1

Needhamov-Schroederov protokol

- Úloha servera S
  - Distribúcia kľúčov
  - Nemusí byť plne online, keďže po vykonaní protokolu, A ani B už nepotrebujú S
- Cvičenie: modifikuje protokol tak, aby využíval asymetrické šifrovanie
  - Aké to bude mať výhody?
- Needhamov-Schroederov protokol sa v súčasnosti neodporúča používať
  - Alternatíva: Kerberos protokol – veľmi rozšírený

Útoky na autentizačné protokoly

Needhamov-Schroederov protokol

- Predpokladajme, že kľúč medzi A, B bol kompromitovaný
  1. A → S: A, B, r_A
  2. S → A: $E_{K_{AS}}(r_A, B, K_{AB}, E_{K_{BS}}(K_{AB}, A))$
  3. Z(A) → B: $E_{K_{BS}}(K_{AB}, A)$
  4. B → Z(A): $E_{K_{AB}}(r_B)$
  5. Z(A) → B: $E_{K_{AB}}(r_B\text{-}1)$
- Z sa úspešne autentizovalo ako A

Útoky na autentizačné protokoly

- Paralelný beh protokolov
  - Odpočuté / prijaté správy v jednom protokole využijeme pri paralelnom behu druhého protokolu
    - Postal-chess problem
- “Chosen text” útok
  - Útočník si volí hodnoty parametrov tak, aby jednoduchšie odhalil informácie o tajnom kľúči
  - CPA / CCA útok na šifrovaciu schému
- „Forced delay“ útok
  - Útočník odpočuje správu (zvyčajne obsahujúcu sekvenčné číslo) a použije ju neskôr

Protokoly na I & A

- Cieľ: Dokázať (vzájomnú) identitu
  - Počas behu protokolu, dokazovateľ nesmie odhaliť svoje tajomstvo útočníkovi
- Fixné heslá
  - Ak je heslo posielané v otvorenom tvare, útočník ho odpočuje
  - Ak je heslo posielané šifrovane, útočník ho môže zopakovať
- Challenge-response protokoly
  - Zabraňujú útokom opakovaním s využitím časovo závislých parametrov
  - Útočník však môže získať nejakú informáciu o tajomstve
    - „Chosen-text“ útoky,
  - Overovateľ môže poznať tajomstvo

Zero Knowledge protokoly

- Dokazovateľ dokáže overovateľovi znalosť tajomstva bez toho, aby odhalil akúkoľvek informáciu o tajomstve
- Postavené na interaktívnych dôkazoch:
  - Pravdepodobnostná verzia „dôkazu“
  - Úlohou dokazovateľa je presvedčiť overovateľa o pravdivosti nejakého tvrdenia cez výmenu správ
- Interaktívne dôkazy na autentizáciu
  - Dôkaz znalosti nejakého tajomstva na základe odpovedania na otázky, pričom správne odpovede vyžadujú znalosť tohto tajomstva

Interaktívne dôkazy

- Úplnosť: interaktívny dôkaz je úplný:
  - Ak sú obidve strany čestné, dôkaz (protokol) skončí úspešne s veľkou pravdepodobnosťou
- Korektnosť: interaktívny dôkaz je korektný, ak existuje efektívny algoritmus M, ktorý
  - Ak je útočník schopný úspešne prebehnúť protokol (presvedčiť overovateľa),
  - potom M môže byť použité na extrakciu informácie z daného útočníka, ktorá môže byť použitá na ďalšie úspešné absolvovanie protokolu
    - Inak povedané, M pozná to tajomstvo

Zero-knowledge protokoly

- Protokol ma vlastnosť „zero-knowledge“ ak
  - Existuje efektívny algoritmus - „simulátor“ S, ktorý
    - dostane na vstupe tvrdenie, ktoré má dokázať
    - bez interakcie s dokazovateľom je schopný generovať transcript neodlíšiteľný od skutočného behu protokolu
- T.j. dokazovateľ neodhalí žiadnu informáciu o svojom tajomstve, okrem tej, ktorá je vypočítateľná z verejne dostupných údajov
  - Aj keď komunikuje s nečestným overovateľom

Zero-knowledge protokoly

V porovnaní s ostatnými protokolmi na I & A:

- Dlhodobé opakovanie protokolu neznižuje bezpečnosť
  - Odolnosť vočí „chosen-text“ útokom
- Nevyžadujú šifrovanie (politické dôvody)
- Zvyčajne menej efektívne
  - Väčšia komunikačná aj výpočtová náročnosť
- Postavené na nedokázaných predpokladoch
  - Podobne ako v prípade asymetrických techník, napr. problém faktorizácie
- „Asymptotické“ dôkazy ZK vlastnosti

Fiatov-Shamirov protokol

- Postavený na probléme počítania odmocnín modulo veľké n = p.q
  - Ekvivaletné problému faktorizácie
- Setup:
  - Dôveryhodný server T vyberie *n = p.q*, prvočísla *p* aj *q* ostávajú utajené.
  - Každý dokazovateľ A si vyberie tajomstvo *s = 1…n-1,* ktoré je nesúdeliteľné s *n*
    - Vypočíta $v = s^2 \pmod n$
    - *v* je verejný kľúč, A ho registruje na serveri T

Fiatov-Shamirov protokol

- Komunikácia počas behu protokolu (A dokazovateľ, B overovateľ)
  - A → B: $x = r^2 \bmod n$, kde r je náhodné $1 \leq r \leq n-1$
  - B → A: náhodný bit e
  - A → B: $y = r * s^e \pmod n$
- B zamietne, ak y = 0, inak
  - akceptuje, ak $y^2 = x.v^e \pmod n$

Fiatov-Shamirov protokol

- Pozorovanie:
  - Útočník C nepozná s, pravdepodobnosť, že odpovie správne je ½ (keďže nevie počítať odmocniny modulo n)
- Pozorovanie:
  - C môže zvoliť $x = r^2/v \pmod n$, vtedy dokáže odpovedať správne pre e=1
  - Pre e = 0 musí poznať odmocinu z x
  - Pravdepodobnosť úspechu ½
- t-násobným opakovaním protokolu dosiahneme pravdepodobnosť podvádzania 2^{-t}

Fiatov-Shamirov protokol

- Odhalená informácia o tajnom kľúči
  - $y = r \pmod n$ – žiadna informácia o s
  - $y = rs \pmod n$ – žiadna informácia o s, keďže r je náhodné a neznáme pre B

Záver

Rôzne autentizačné schémy

- Fixné hesla
  - Jednoduchý útok opakovaním
  - Slovníkový útok
- Jednoduché challenge-response protokoly
  - Poskytujú ochranu voči útokom opakovaním
  - Vyžadujú zdieľané tajomstvo, resp. dôveryhodnú distribúciu verejných kľúčov
- Key distribution center
  - Využitie dôveryhodnej 3. strany na distribúciu kľúča
  - Užitočné v prípade veľkého množstva komunikujúcich strán
- Zero-knowledge protokoly
  - Neposkytujú žiadnu informáciu o tajomstve.
  - Nie je potrebné pamätať si tajné kľúče na serveri