

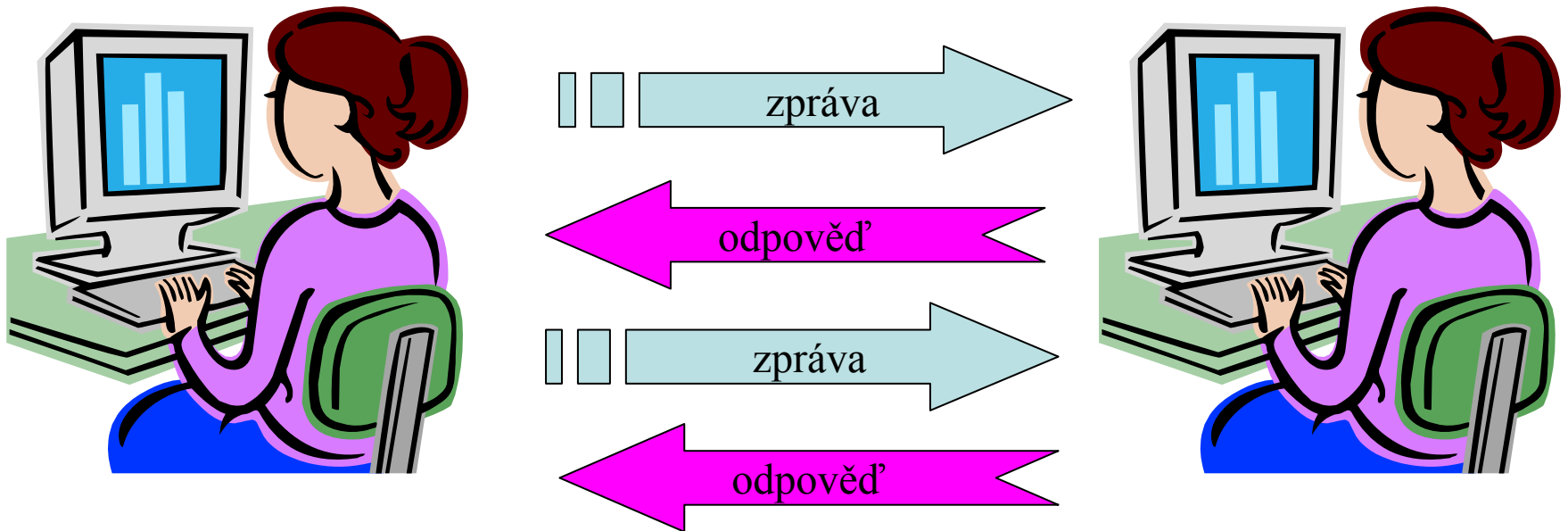
# PV157 – Autentizace a řízení přístupu



## Autentizační protokoly

# Protokol

- Protokol je několikastranný algoritmus definovaný posloupností kroků, které specifikují akce prováděné dvěma a více stranami, pro dosažení určitého cíle



# Kryptografické protokoly

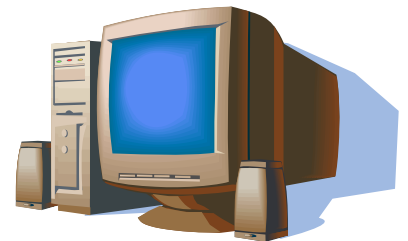
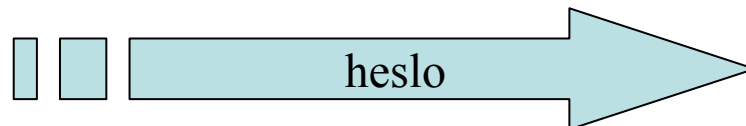
- **Autentizační protokol** – zajistí jedné straně určitou míru jistoty o identitě jiné strany (té, se kterou komunikuje), příp. protokol oboustranný
- **Protokol pro ustavení klíče** (key establishment protocol) – ustaví sdílené tajemství (typicky klíč)
- **Autentizovaný protokol pro ustavení klíče** (authenticated key establishment protocol) – ustaví sdílené tajemství se stranou, jejíž identita byla potvrzena

# Autentizační protokoly

- Během protokolu autentizujeme:
  - Pouze jednu ze stran
  - Obě strany
  - Kontinuální autentizace
- Kdo koho autentizuje
  - Alice vyzývá Boba, aby se autentizoval
  - Bob se autentizuje rovnou sám bez výzvy

# Autentizace heslem

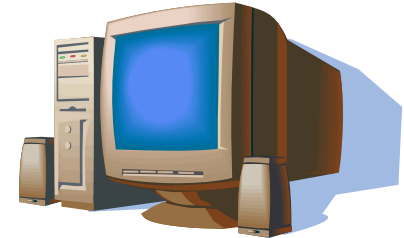
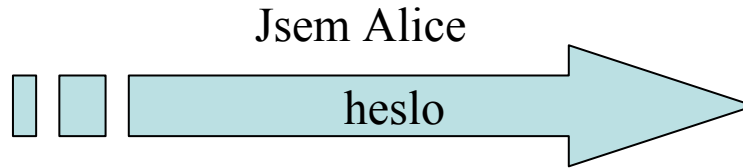
- Alice se autentizuje Bobovi tak, že mu pošle své heslo
- Heslo je možné odposlechnout
- Bob po úspěšné Alicině autentizaci zná Alicino heslo a může se (např. vůči Cyrilovi) autentizovat jako Alice (pokud Alice používá stejné heslo pro autentizace vůči různým stranám)



# Útok impersonací (vydáváním se za jiného)



Alice



Bob



Emil je pasivní útočník (odposlouchává komunikaci)

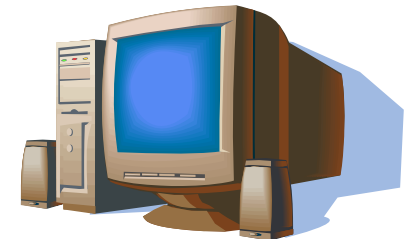
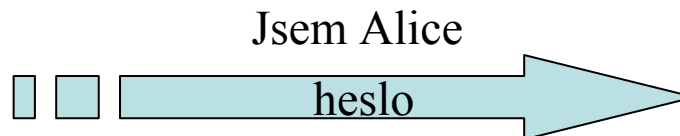
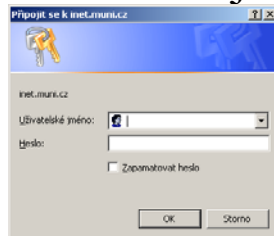


Emil

Emil zadá heslo do existující aplikace

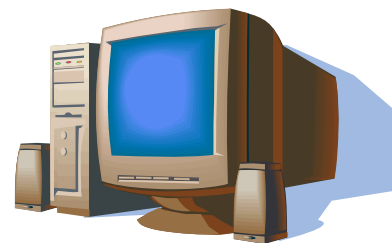
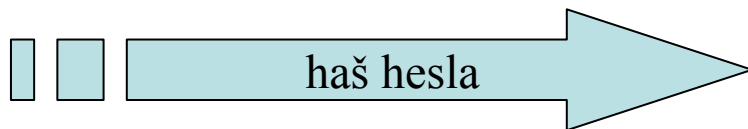
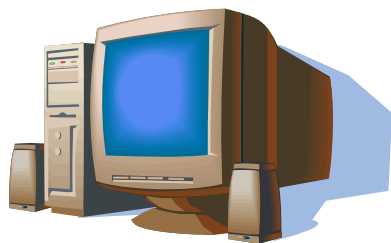


Emil



# Hašované heslo

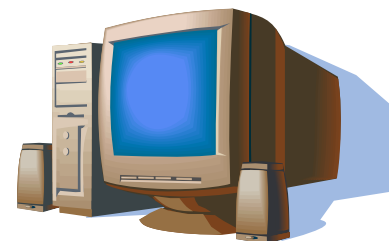
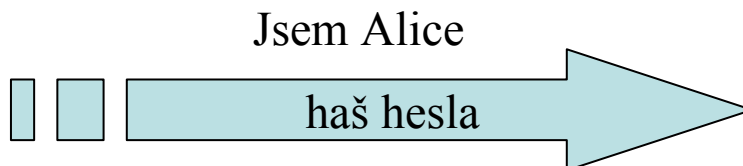
- Při autentizaci se neposílá heslo samotné, ale pouze haš hesla
- Kdo odposlechne haš nezíská automaticky heslo
- Haš však lze použít pro podvodnou autentizaci



# Útok přehráním



Alice



Bob



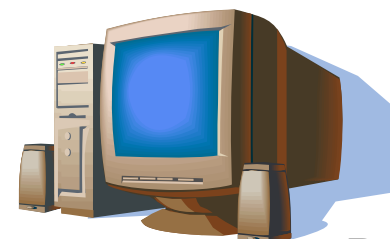
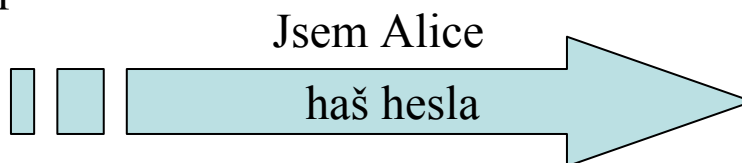
Emil je pasivní útočník (odposlouchává komunikaci)

Emil

Emil nezná heslo, ale pošle odposlechnutý haš hesla, pomocí své pomocné aplikace



Emil



Bob

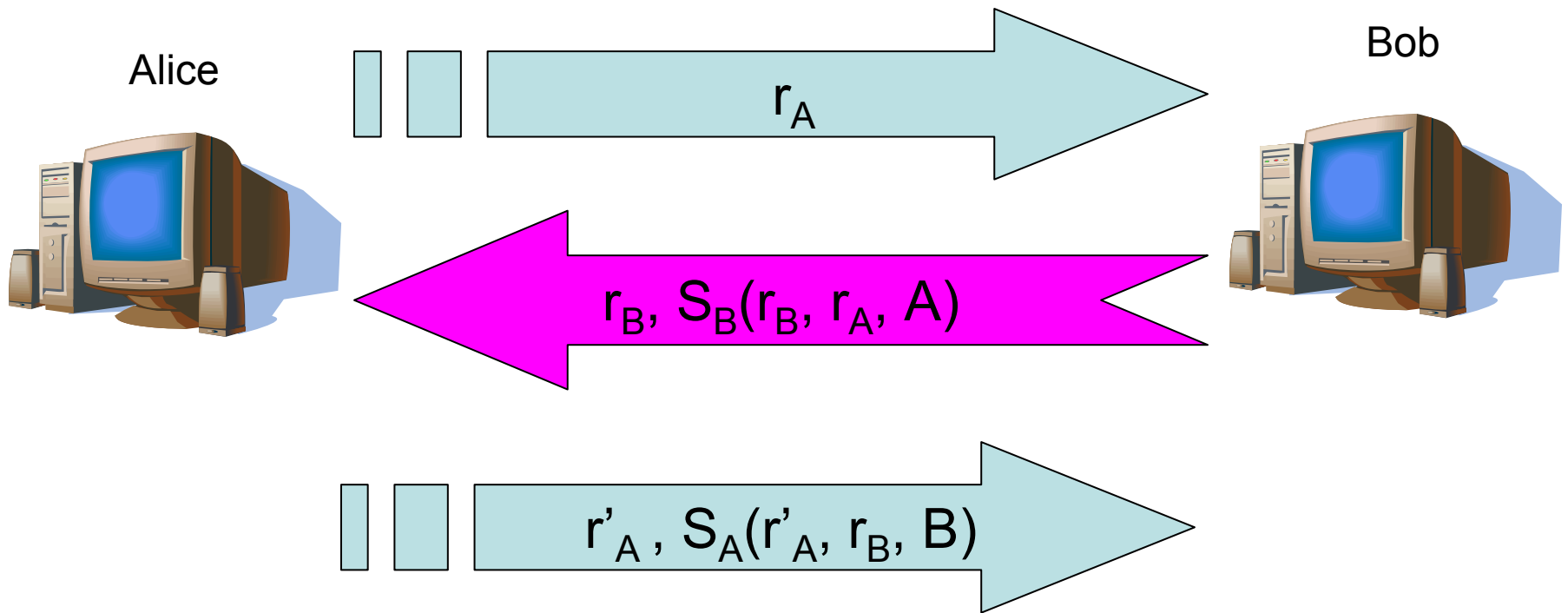


# Další útoky na protokoly

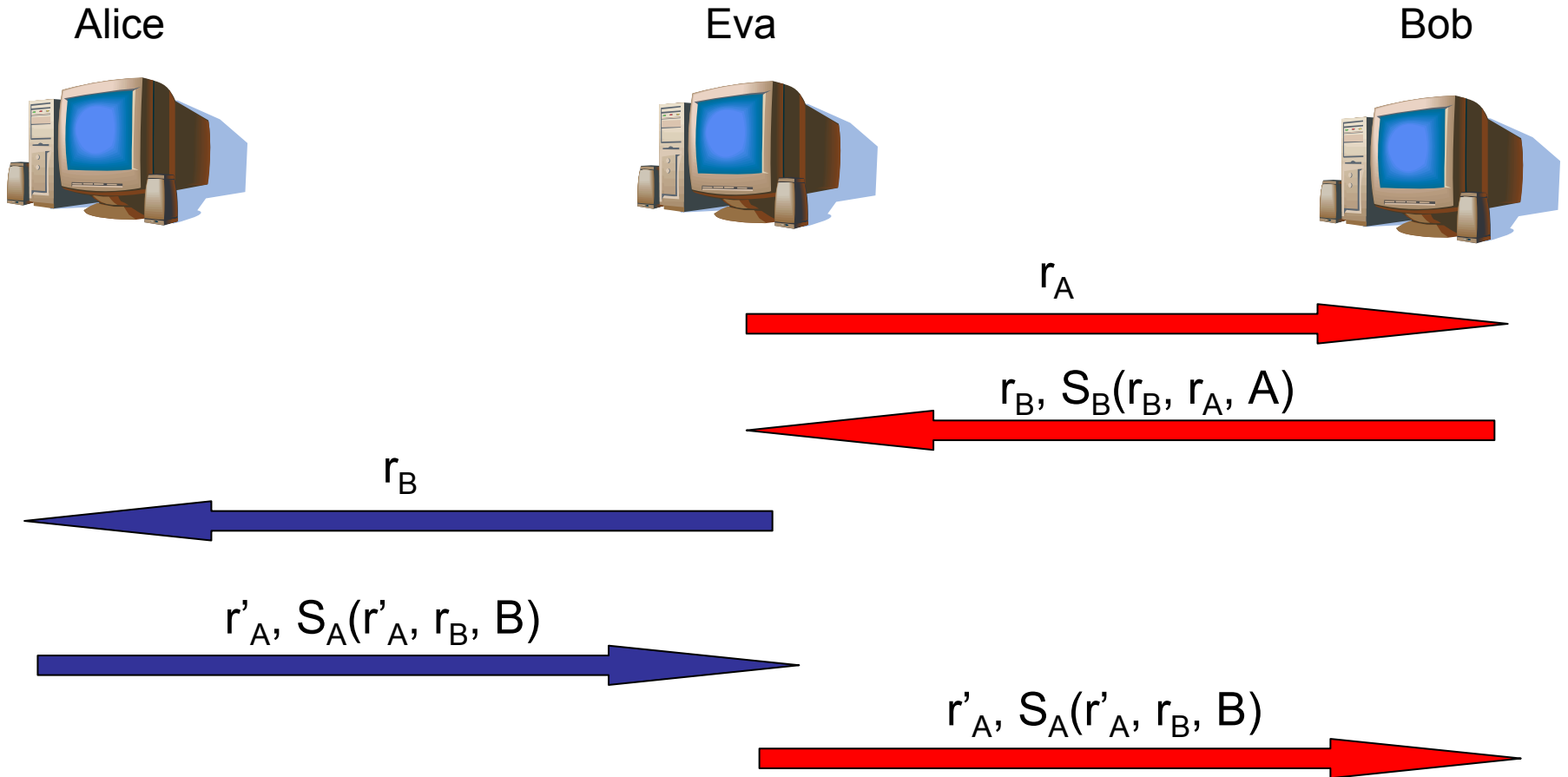
- Zmíněné útoky impersonací a přehráním
- Úplný výčet je nesnadný, ale zmínit je třeba
  - útoky **prolínáním** (interleaving) – kombinujeme zprávy z více průběhů – obvykle, ale ne nutně jen, stejného protokolu – ať již ukončených, nebo právě probíhajících (viz další slajd)
  - **slovníkové** útoky – na protokoly využívající hesla, diskutováno u autentizace uživatelů
  - útoky **využitím známého klíče** (known-key) – obvyklé u protokolů pro ustanovení klíče, kde se klíč ustanoví na základě staršího/ch (útočníkovi známého/ch) klíče/ů
  - další později nebo příště...

# Útok prolínáním (1)

- Mějme autentizační protokol:



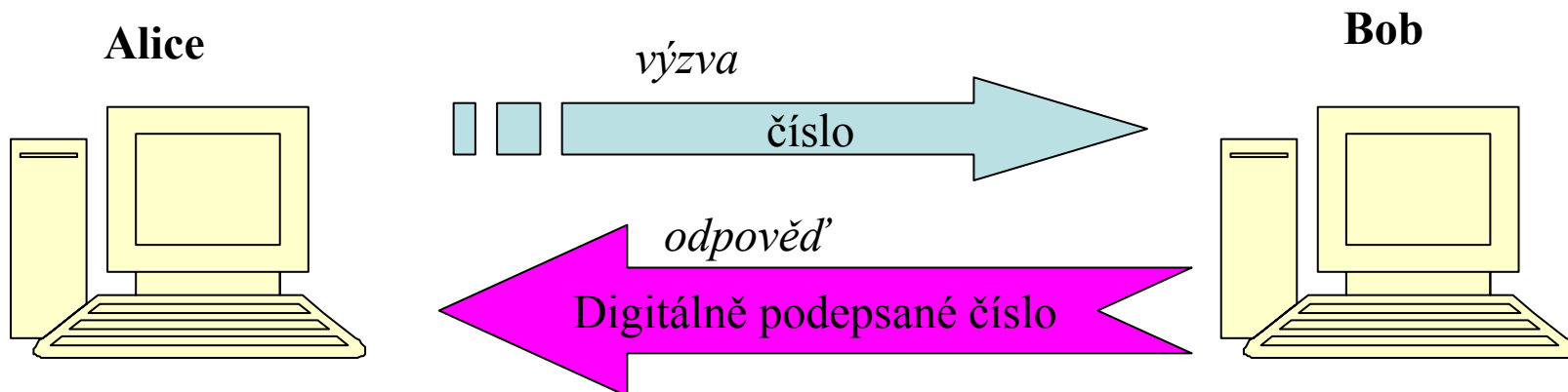
# Útok prolínáním (2)



Evě se podařilo vydávat se vůči Bobovi za Alici

# Protokoly výzva-odpověď

- Protokoly typu výzva-odpověď (challenge-response)
  - Odposlechem výzvy i odpovědi útočník moc nezíská
  - Bob se může přesvědčit o identitě Alice, bez získání jejího tajemství



# Časově proměnné parametry

- **Náhodná čísla** (random numbers) – čísla, která jsou nepredikovatelná (v tomto kontextu zahrnujeme pod náhodná čísla i čísla pseudonáhodná). Použitím náhodných čísel zajišťujeme jedinečnost a „aktuálnost/čerstvost“. Získat skutečně náhodná čísla je netriviální (vyžaduje speciální HW zařízení). V praxi obvykle používáme pseudonáhodná čísla (které na základě tajného stavu - semínka (seed) generují sekvence čísel). Značíme **r**.
- **Sekvence** (sequence numbers) – monotonně rostoucí posloupnost čísel (obě strany musí dlouhodobě uchovávat informaci o poslední hodnotě). Jednoznačně identifikují zprávy a umožňují detekovat útoky přehráním předchozí komunikace. Značíme **n**.
- **Časová razítka** (timestamps) – obě strany musí synchronizovat a zabezpečit hodiny. Zajišťují jedinečnost a časovou přesnost. Značíme **t**.

# Protokoly výzva-odpověď

- Založené na symetrických technikách
  - Symetrické šifrování
  - Jednosměrná funkce s klíčem
  - Generátory passcode
- Založené na asymetrických technikách
  - Dešifrování
  - Digitální podpis

# Symetrické techniky

- Založené na symetrickém šifrování (Alice a Bob sdílí tajný symetrický klíč **K**)
- Standard ISO/IEC 9798-2
- Jednostranná autentizace (časové razítko)
  - $A \rightarrow B: E_K(t_A, "B")$
- Možné útoky
  - Útok přehráním: odposlechnu  $E_K(t_A, "B")$  a pošlu jej rychle znovu (v době platnosti  $t_A$ )
  - Změna hodin: odposlechnu  $E_K(t_A, "B")$ , později změním hodiny B tak, aby odpovídaly času  $t_A$  a znovu pošlu  $E_K(t_A, "B")$

# Symetrické techniky

- Jednostranná autentizace (náhodné číslo)
  - $A \leftarrow B: r_B$
  - $A \rightarrow B: E_K(r_B, "B")$
- Možné útoky
  - Útočník odposlouchává a ukládá  $[r_B, E_K(r_B, "B")]$ , pokud se výzva  $r_B$  opakuje, pak je schopen poslat správnou odpověď. Případně se může aktivně snažit ovlivnit vytváření náhodných  $r_B$  (např. ovlivněním vstupu generátoru náhodných čísel Boba).
- Oboustranná autentizace (náhodná čísla)
  - $A \leftarrow B: r_B$
  - $A \rightarrow B: E_K(r_A, r_B, "B")$
  - $A \leftarrow B: E_K(r_B, r_A)$

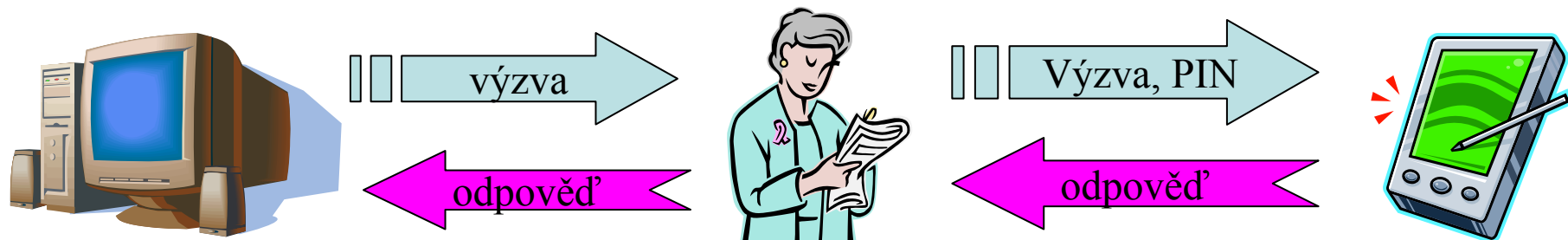


# Symetrické techniky

- Založené na klíčovaných jednosměrných funkcích (Alice a Bob sdílí tajný symetrický klíč **K**)
- Standard ISO/IEC 9798-4, protokoly SKID
- Oboustranná autentizace
  - $A \leftarrow B: r_B$
  - $A \rightarrow B: r_A, h_K(r_A, r_B, "B")$
  - $A \leftarrow B: h_K(r_B, r_A, "A")$
  - $h_K$  je MAC algoritmus

# Symetrické techniky

- Generátory passcode – hand-held (PDA, kapesní počítače) pro bezpečné uložení dlouhodobých klíčů doplněné zadáním PINu uživatele
- Subjekty A, B sdílí tajný klíč  $s_A$  a tajný PIN  $p_A$ 
  - $A \leftarrow B: r_B$
  - subjekt A zadá do generátoru přijatou výzvu  $r_B$  a vloží svůj PIN  $p_A$
  - $A \rightarrow B: f(r_B, s_A, p_A)$



# Asymetrické techniky

- Založené na dešifrování soukromým klíčem
- Jednostranná autentizace
  - $A \leftarrow B: h(r), \text{„B“}, P_A(r, \text{„B“})$
  - $A \rightarrow B: r$
- $h$  – hašovací funkce
- $h(r)$  slouží k prokázání znalosti  $r$  bez jeho odhalení

# Asymetrické techniky

- Založené na digitálním podpisu
- Standard ISO/IEC 9798-3
- Jednostranná autentizace (časové razítko)
  - $A \rightarrow B: cert_A, t_A, "B", S_A(t_A, "B")$
- Možné útoky
  - Útok přehráním: odposlechnu  $S_A(t_A, "B")$  a pošlu jej rychle znovu (v době platnosti  $t_A$ )
  - Změna hodin: odposlechnu  $S_A(t_A, "B")$ , později změním hodiny B tak, aby odpovídaly času  $t_A$  a znovu pošlu  $S_A(t_A, "B")$

# Asymetrické techniky

- Jednostranná autentizace (náhodné číslo)
  - $A \leftarrow B: r_B$
  - $A \rightarrow B: cert_A, r_A, \text{"B"}, S_A(r_A, r_B, \text{"B"})$
  - $r_A$  zde zabraňuje útokům s vybraným textem
- Možné útoky
  - Obdobné útoky na náhodné  $r_B$  jako v případě symetrických technik
- Oboustranná autentizace (náhodná čísla)
  - $A \leftarrow B: r_B$
  - $A \rightarrow B: cert_A, r_A, \text{"B"}, S_A(r_A, r_B, \text{"B"})$
  - $A \leftarrow B: cert_B, \text{"A"}, S_B(r_B, r_A, \text{"A"})$

# Protokoly pro správu klíčů

- Účel
  - Přenos klíče
  - Ustavení klíče
  - Aktualizace klíče (strany sdílí dlouhodobý klíč **K**)
  - Zároveň i autentizace jedné nebo obou stran
- Počet stran
  - Protokol pro dvě strany
  - Protokol s důvěryhodnou třetí stranou

# Symetrické techniky přenosu klíče

- Aktualizace klíče založená na symetrické šifře (Alice a Bob sdílí tajný klíč  $K$ )
  - Přenos klíče (1 zpráva, časové razítko)
    - $A \rightarrow B: E_K(r_A, t_A, "B")$
  - Přenos klíče (výzva-odpověď, náhodné nebo sekvenční číslo)
    - $A \leftarrow B: n_B$
    - $A \rightarrow B: E_K(r_A, n_B, "B")$

# Symetrické techniky přenosu klíče

- Přenos klíče odvozením
  - $A \rightarrow B: r_A$
  - Nový klíč  $W = E_K(r_A)$
- Aktualizace klíče se vzájemnou autentizací
  - AKEP2 (Authenticated Key Exchange Protocol 2)
  - $A \rightarrow B: r_A$
  - $A \leftarrow B: ("B", "A", r_A, r_B), h_K("B", "A", r_A, r_B)$
  - $A \rightarrow B: ("A", r_B), h_K("A", r_B)$
  - Nový klíč  $W = h'_{K'}(r_B)$
  - $h_K$  je MAC algoritmus,  $h'$  je MAC algoritmus (odlišný od  $h$ ), obě strany sdílí  $K$ , z  $K$  je odvozen  $K'$

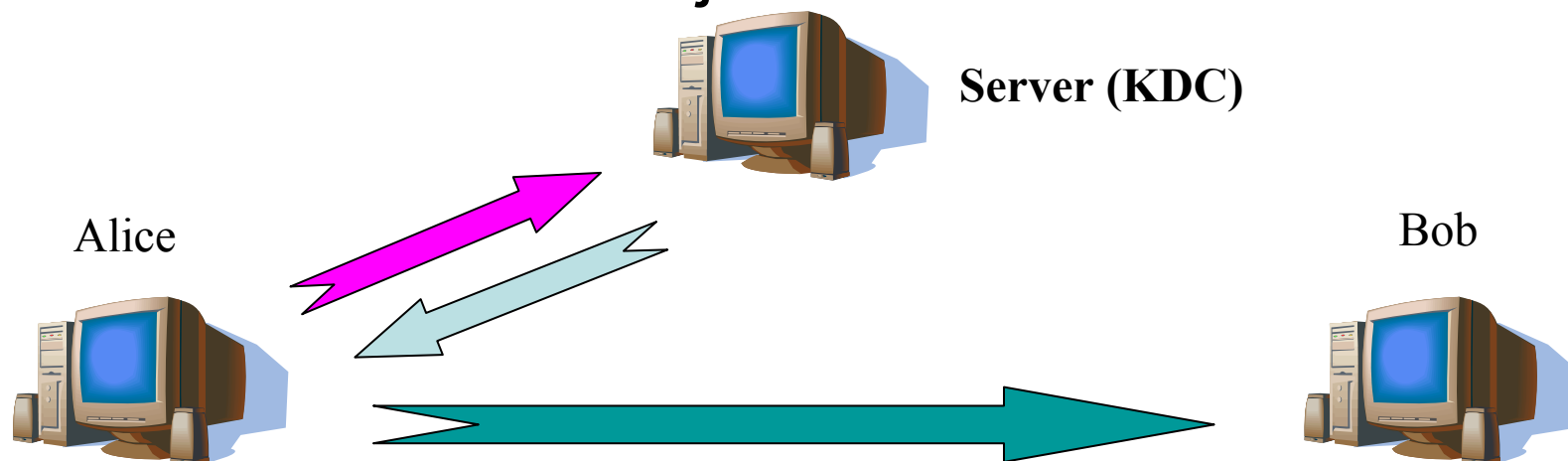


# Protokol bez klíčů

- Přenos klíče bez předchozího sdíleného tajemství
  - Shamirův protokol bez klíčů (Shamir's no-key protocol)
  - Komutativní šifra  $E$
  - Každá strana má svůj klíč  $K_A, K_B$
  - $A \rightarrow B: E_{K_A}(X)$
  - $A \leftarrow B: E_{K_B}(E_{K_A}(X))$
  - $A \rightarrow B: E_{K_B}(X)$
  - Nyní obě strany sdílí  $X$ ; byly nutné 3 zprávy

# Kerberos

- **KDC** (key distribution center) – server sdílí klíč s každým klientem; (klienti však mezi sebou klíče nesdílí); server distribuuje klíče, které generuje.
- **KTC** (key translation center) – server negeneruje klíče sám; klíč dodá jedna ze stran; server klíč distribuuje



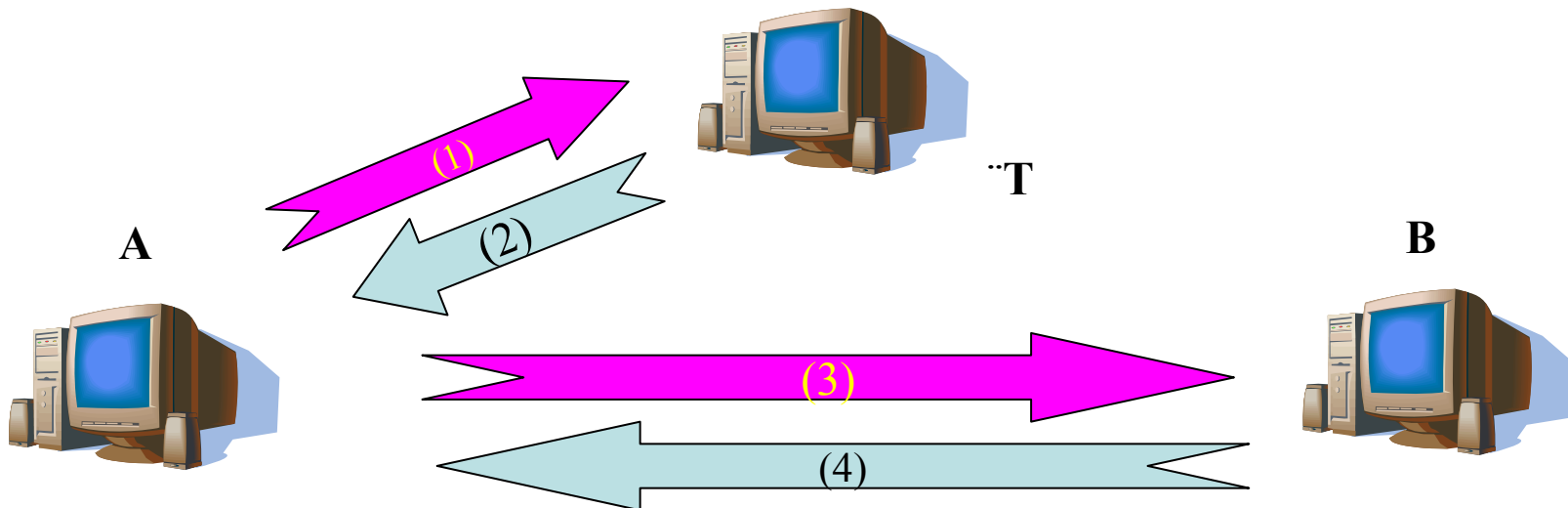
# Kerberos

- Vznikl při projektu Athena na MIT
- Symetrická šifra E
- 2 strany (A, B) a důvěryhodný autentizační server (značíme T)
- Cíl:
  - autentizace subjektu A vůči B
  - ustavení klíče k (zvolí T)
  - případně distribuce tajemství sdíleného A a B
- Každá strana sdílí tajemství se serverem  $K_{AT}$ ,  $K_{BT}$



# Kerberos

- Zjednodušená verze protokolu
  - L – doba platnosti („lifetime“)
  - Def.:  $\text{ticket}_B = E_{K_{BT}}(k, \text{"A"}, L)$ ,  $\text{auth} = E_k(\text{"A"}, T_A)$
  - (1)  $A \rightarrow T: \text{"A"}, \text{"B"}, n_A$
  - (2)  $A \leftarrow T: \text{ticket}_B, E_{K_{AT}}(k, n_A, L, \text{"B"})$
  - (3)  $A \rightarrow B: \text{ticket}_B, \text{auth}$
  - (4)  $A \leftarrow B: E_k(T_A)$



# Asymetrické techniky přenosu klíče

- Zašifrování podepsaných klíčů
  - $A \rightarrow B: P_B(S_A("B", k, t_A))$
  - (volitelné) časové razítko  $t_A$  zároveň autentizuje A vůči B
  - Pouze v případě, kdy z podpisu lze získat podepsaná data
- Separátní šifrování a podpis
  - $A \rightarrow B: P_B(k, t_A), S_A("B", k, t_A)$
  - Pouze v případě, kdy z podpisu nelze získat podepsaná data
- Podepsání zašifrovaných klíčů
  - $A \rightarrow B: t_A, P_B("A", k), S_A("B", t_A, P_B("A", k))$

# Asymetrické techniky přenosu klíče

- X.509 obousměrná autentizace s přenosem klíče
- Def.:  $D_A = (t_A, r_A, \text{"B"}, P_B(k_1))$   
 $D_B = (t_B, r_B, \text{"A"}, P_A(k_2))$
- Protokol
  - $A \rightarrow B: cert_A, D_A, S_A(D_A)$
  - $A \leftarrow B: cert_B, D_B, S_B(D_B)$

# Asymetrické techniky ustavení klíče

- Diffie-Hellman protokol pro ustavení sdíleného tajemství
  - Společné prvočíslo  $p$ , generátor  $\alpha$  v  $Z_p$
  - A volí tajné  $x$ , B volí tajné  $y$
  - $A \rightarrow B: \alpha^x \bmod p$
  - $A \leftarrow B: \alpha^y \bmod p$
  - A a B sdílí  $K = \alpha^{xy} \bmod p$

# Zero-knowledge protokoly

- Český překlad: protokoly s nulovým rozšířením znalostí
- Jdou dále než protokoly sdělující hesla i protokoly typu výzva-odpověď
- Zero-knowledge – umožňují demonstrovat znalost nějakého tajemství bez odhalení jakékoliv informace použitelné pro získání tajemství
- Úplnost (completeness) – poctivé strany vždy dosáhnou úspěšného výsledku
- Korektnost (soundness) – pravděpodobnost, že nepoctivý útočník se může úspěšně vydávat za jinou stranu je mizivá



# Protokol Fiat-Shamir

- Důvěryhodná strana T volí modulus  $n = p \cdot q$  (jako v RSA),  $n$  zveřejní, ale  $p$  a  $q$  uchová v tajnosti
- A volí tajné  $s$  (nesoudělné s  $n$ ,  $1 \leq s \leq n-1$ ), spočítá  $v = s^2 \bmod n$ . Veřejný klíč A je  $v$ .
- Subjekt A se autentizuje subjektu B:
  - $A \rightarrow B: x = r^2 \bmod n$
  - $A \leftarrow B: e = 0$  nebo  $1$
  - $A \rightarrow B: y = r \cdot s^e \bmod n$  (B ověří zda  $y^2 = x \cdot v^e \pmod n$ )
  - Podvést by šlo:
    - Volbou  $r$ ,  $x=r^2 \bmod n$ , pro  $e=0$
    - Volbou  $r$ ,  $x=r^2/v$  (přízemž odmocninu z  $x$  neznám), pro  $e=1$
- Opakujeme  $t$ -krát. Pravděpodobnost podvádění je  $2^{-t}$ .

# Protokoly vyšší úrovně – SSL/TLS

Protokol SSL/TLS poskytuje:

- Autentizaci stran – strany jsou autentizovány pomocí certifikátů a protokolu výzva-odpověď
- Integritu – autentizační kódy (message authentication code - MAC) zajišťují integritu a autenticitu dat
- Důvěrnost – po úvodní inicializaci („handshake“), je ustaven symetrický šifrovací klíč, kterým je šifrována všechna následující komunikace (včetně přenosu hesel apod.)

# Principy SSL/TLS

- Pozice SSL/TLS
  - Mezi aplikační vrstvou a protokolem TCP
  - SSL/TLS nevidí do aplikačních dat
  - SSL/TLS neprovádí elektronické podepisování přenášených dat

Aplikační vrstva
<b>SSL/TLS</b>
TCP/UDP
IP
Linková vrstva
Fyzická vrstva

# Komponenty SSL/TLS

- Složení protokolu SSL/TLS z komponent
  - Record Layer Protocol – zpracovává aplikační data
  - Handshake Protocol – úvodní domluva parametrů
  - Change Cipher Specification Protocol – použití nových parametrů šifrování
  - Alert protocol – informace o chybách a varováních

# Klíče v SSL/TLS

- Použití klíčů
  - Klient generuje PreMasterSecret, šifruje veřejným klíčem serveru a posílá serveru
  - Obě strany vytvoří blok klíčů z PreMasterSecret (posílá se šifrovaně) a náhodných čísel ClientHello a ServerHello (posílají se nešifrovaně)
  - Blok klíčů tvoří klíče pro
    - MAC klient → server
    - MAC server → klient
    - šifrování klient → server
    - šifrování server → klient
    - inicializační vektory

# Record Layer Protocol

- Základní vrstva protokolu
- Pracuje nad TCP/IP (nebo jiným transportním protokolem).
- Umožňuje kombinaci s různými protokoly vyšší úrovně (HTTP, FTP, telnet apod.), které běží beze změny
- Posloupnost kroků
  - rozdělení dat na bloky o max. velikosti  $2^{14}$  bajtů
  - komprimace dat
  - výpočet MAC
  - doplnění na délku bloku šifrovacího algoritmu
  - šifrování

# Inicializační fáze

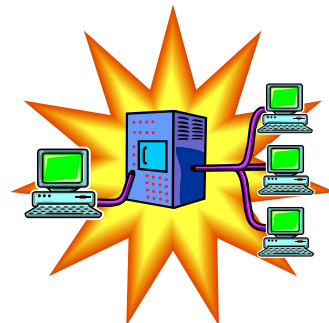
- Handshake Protocol
  - Umožňuje vzájemnou autentizaci serveru a klienta
  - Implicitně je autentizace serveru povinná a autentizace klienta volitelná
  - Autentizace prezentací **certifikátů veřejných klíčů** a znalostí odpovídajících soukromých klíčů
  - Během inicializační fáze jsou vyměněna náhodná čísla a další data, nutná pro výpočet bloku klíčů

# SSL/TLS

**Client**



**Server**



Client Hello



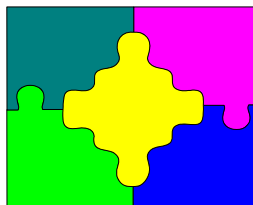
Server Hello, ( , Client Cert Request, ... )



Client Key Exchange, Cipher Spec, ( , ... )



Application



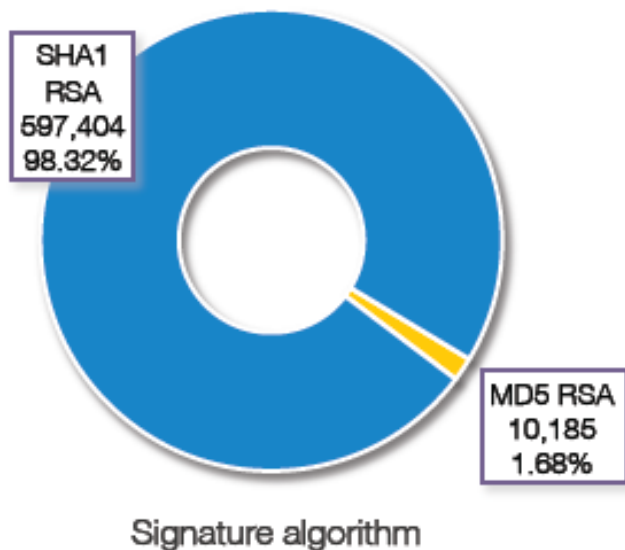
Data



**S E C U R E**



# SSL/TLS v praxi



Key length	Certificates seen
512	3,005
1024	386,694
2048	211,155
4096	6,315
8192	14
Other	406

- Analýza SSL certifikátů provedená v roce 2010
  - 22,65 milionů web serverů s podporou SSL
  - Jen 720 tisíc serverů s certifikátem se správným jménem

# IPsec

- Protokoly IPv4 – nedostatečná bezpečnost
- Historie
  - Myšlenka IPsec již v roce 1991
  - RFC v roce 1998
  - vývoj neustále pokračuje
  - IPsec pro IPv4 jen přechodné řešení, neboť IPv6 již řeší problémy bezpečnosti
- IPv6
  - Větší množství adres (adresy IPv4 nebudou již brzy stačit)
  - Bezpečnost (IPsec povinný)
  - Mobilita

# IPsec

- IPsec zajišťuje
  - Autentizaci původu dat – každý datagram je ověřován, zda byl odeslán uvedeným odesilatelem
  - Integrita dat – ověřuje se, zda data nebyla při přenosu změněna
  - Důvěrnost dat – data jsou před přenosem šifrována
  - Ochrana před útokem přehráním – útočník nemůže zneužít odposlechnutou komunikaci k útoku přehráním
  - Automatickou správu klíčů

# IPsec – AH

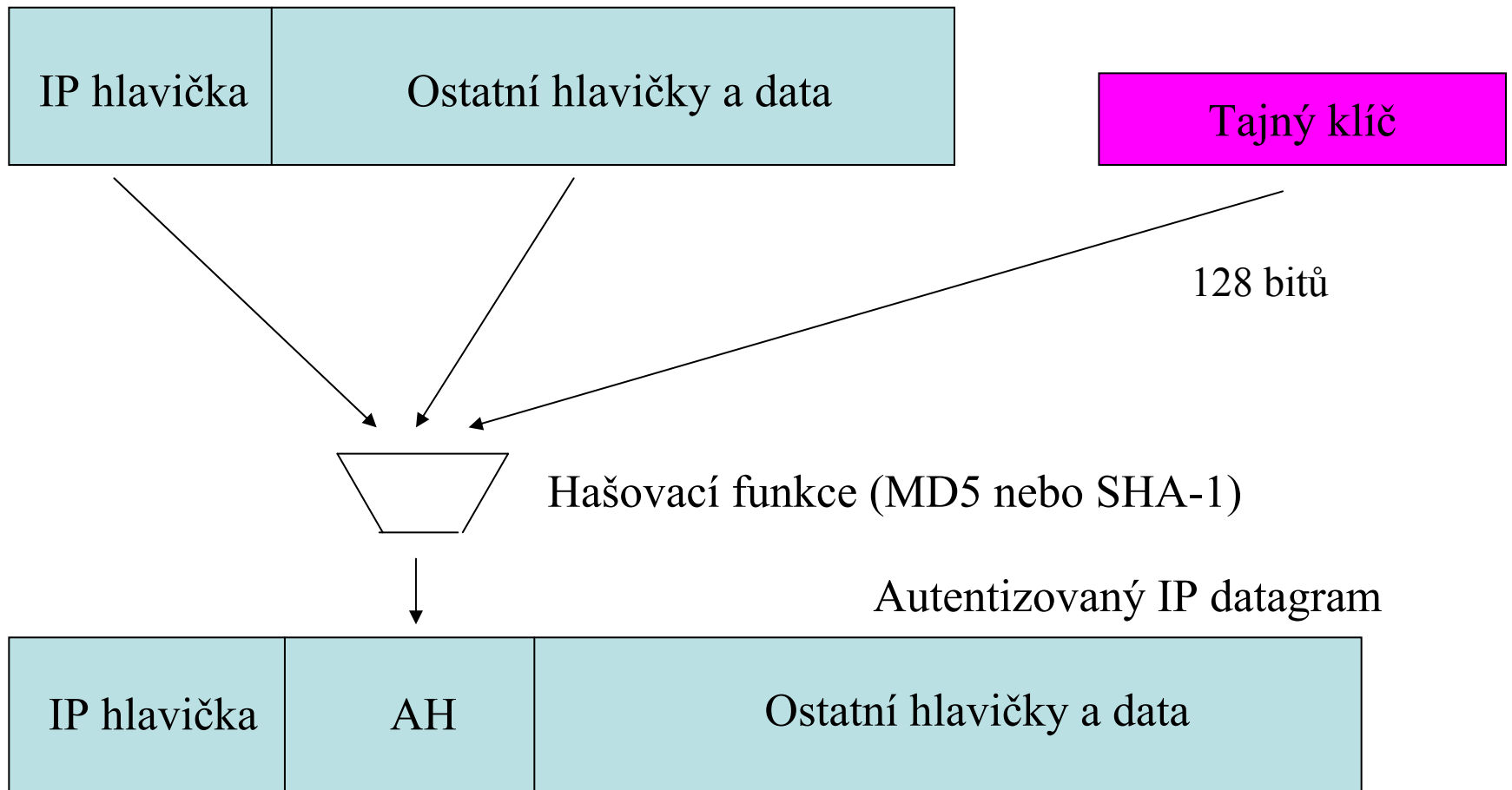
- Autentizační hlavička (AH)

Next header	Length	Reserved
Security Parameter Index		
Sequence number field		
Authentication Data		

- Autentizační hlavička slouží k zajištění původu dat, integrity dat a chrání vůči útoku přehráním. Je použit MAC kombinovaný se sekvenčním číslem.

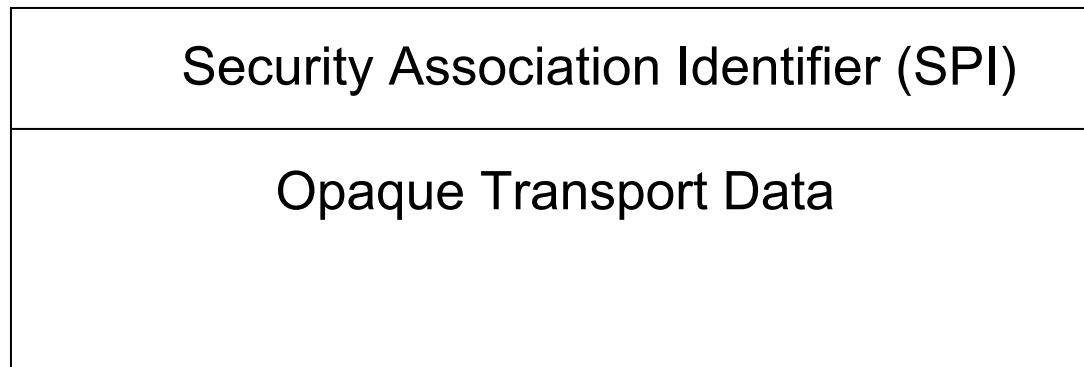
# IPsec - AH

Původní IP datagram



# IPsec - ESP

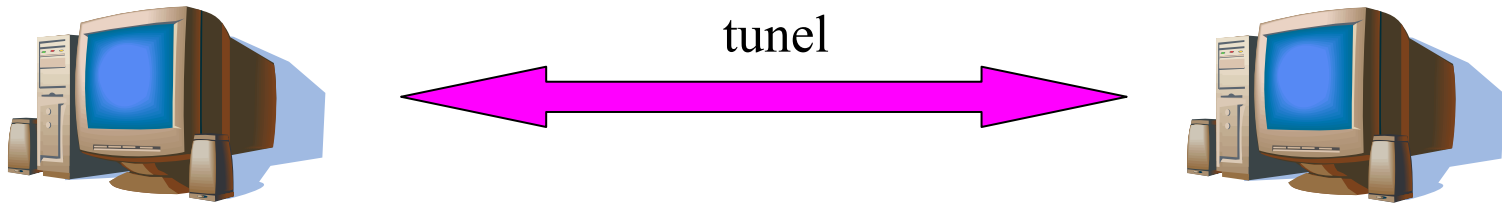
- Encapsulated Security Payload (ESP) header



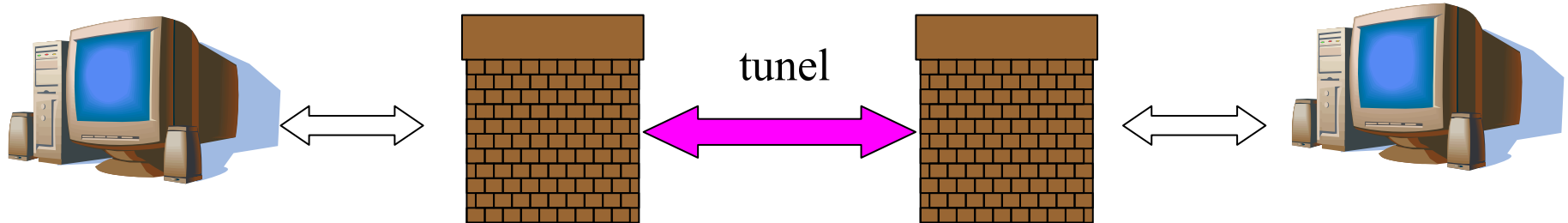
- ESP zajišťuje integritu a autenticitu dat, brání útokům přehráním a zajišťuje **důvěrnost dat**. Je použit symetrický šifrovací klíč sdílený oběma komunikujícími stranami.

# Režimy IPsec

- Transportní režim (end-to-end)

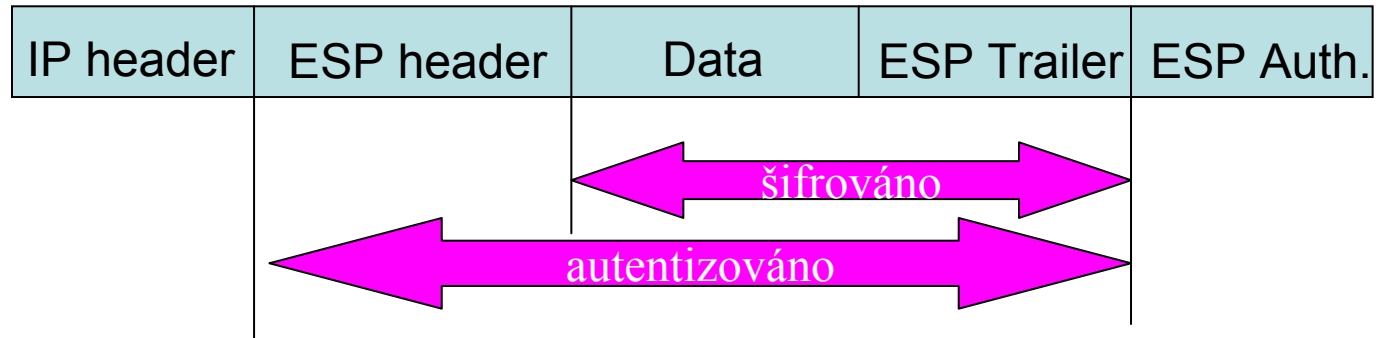


- Tunelovací režim (firewall-to-firewall)

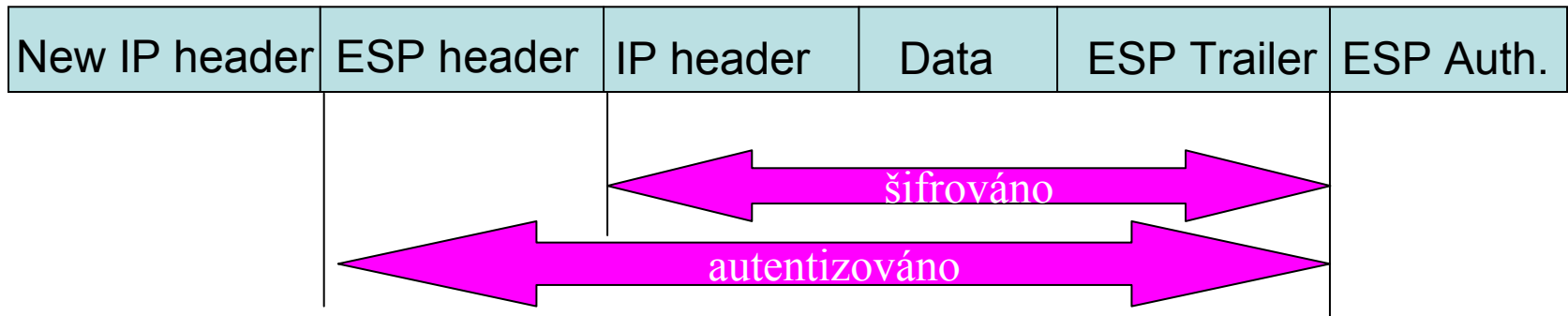


# IPsec

- Standardní IP:
- Režimy provozu IPsec
  - Transportní režim (point-to-point)



- Tunelovací režim





# IPsec – správa klíčů

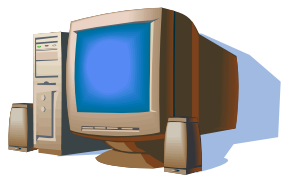
- Oakley
  - protokol pro ustavení společného klíče
  - založen na protokolu Diffie-Hellman, ale:
    - strany jsou autentizovány (brání man-in-the-middle útoku)
      - sdílené klíče, dohodnuté předem
      - Veřejné klíče DNS (viz DNSSEC)
      - RSA klíče podle PGP
      - RSA klíče včetně certifikátu podle X.509
      - DSS klíče včetně certifikátu podle X.509
    - pomocí časově proměnných parametrů se brání útokům přehráním
    - pomocí tzv. cookies se brání útokům typu „DoS“ (prováděné výpočty jsou totiž časově náročné)
    - umožňuje dohodu na použité skupě
- ISAKMP
  - framework (nezávislý na konkrétních šifrovacích algoritmech) pro správu klíčů a bezpečnostních atributů

# Útoky

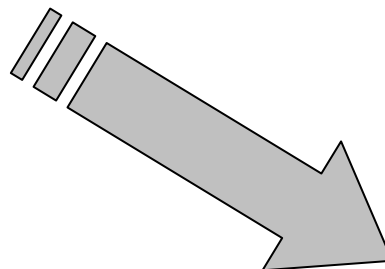
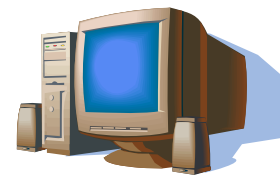
- **Pasivní útočník** – analyzuje odchycená šifrovaná data
- **Aktivní útočník** – modifikuje data a/nebo vytváří nové zprávy
- **Zosobnění** (impersonation) – jedna strana se vydává za stranu jinou
- **Přehrání** (replay attack) – využití dříve poslané informace
- **Odras** (reflection attack) – využití odeslané zprávy k okamžitému poslání odesilateli
- **Volený text** (chosen-text attack) – vhodné volení výzev (v protokolech výzva-odpověď) pro získání dlouhodobého klíče

# Útok přehráním

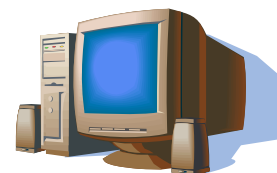
**Alice**



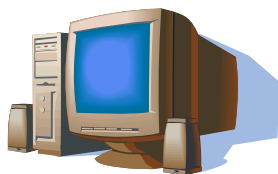
**Bob**



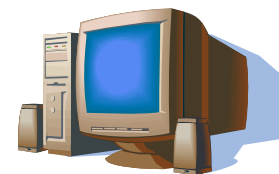
**Eva**



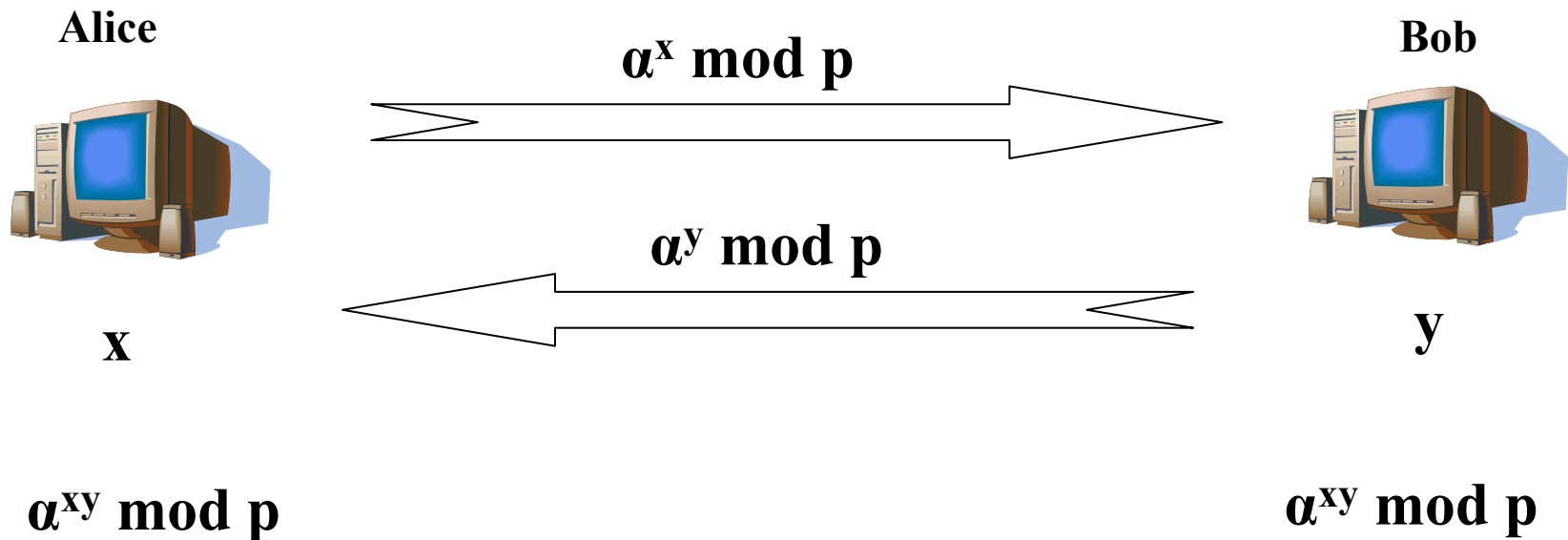
**Eva**



**Bob**



# Protokol Diffie-Hellman (opak.)



# Útok „Man in the middle“

