24 Hesla a bezpečnost. Bezpečnostní politika v organizaci. Matice rizik a mapa rizik. Hesla, bezpečná tvorba a ochrana. Vysvětli pojmy kódování a šifrování, hash, salt, pepper, iterace. Popiš a porovnej symetrické a asymetrické šifry. Základní techniky prolomení hesla a ochrana před nim.

Hesla a bezpečnost

1. **Silná hesla:**

   - Používejte unikátní hesla s alespoň 16 znaky obsahujícími číslice, velká a malá písmena a speciální symboly.
   - Neopakujte stejná hesla pro různé účty.
   - Neukládejte hesla do prohlížeče.
   - Nepoužívat stejné heslo na více účtech/aplikacích

2. **Dvoufázové ověření:**

   - Aktivujte dvoufázové ověření pro dodatečnou ochranu.

3. **Ochrana při připojování k sítím:**

   - Nepřihlašujte se k důležitým účtům na veřejných Wi-Fi sítích.
   - Používejte šifrovaná připojení (např. HTTPS).

4. **Aktualizace a bezpečnostní software:**

   - Aktualizujte operační systémy a prohlížeče.
   - Používejte antivirový software a firewall.

5. **Správa hesel:**

   - Používejte správce hesel pro bezpečné ukládání a generování hesel.
   - Aktualizujte si heslo, nepoužívejte jedno heslo dokola
   - Šifrování hesel

6. **Opatrnost při klikání:**

   - Neklikejte na podezřelé odkazy v e-mailech.
   - Zkontrolujte adresy webových stránek před zadáním hesla.

7. **Zabezpečení zařízení:**

   - Zamkněte zařízení, když je necháte bez dozoru.

- Šifrujte pevné disky a používejte autentizaci.

8. **Ochrana před malwarem:**

   - Stahujte software pouze z důvěryhodných zdrojů.
   - Zkontrolujte soubory na škodlivý software.

9. **Ochrana před phishingem:**

   - Buďte opatrní při zadávání citlivých informací online.

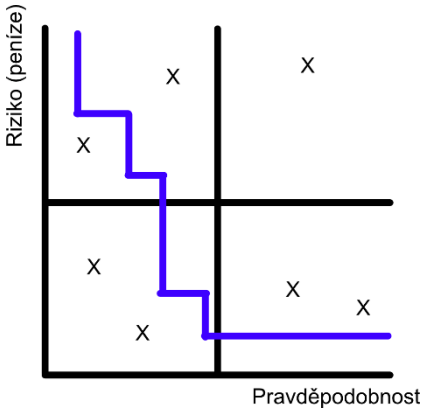
10. **Vzdělávání uživatelů:**

    - Školte uživatele/zaměstnance o hrozbách a jak jim zabránit

Matice rizik a mapa rizik

Matice rizik:

- Matice rizik je tabulkový nástroj, který uspořádává rizika na základě jejich pravděpodobnosti a dopadu.
- Používá se k hodnocení a prioritizaci rizik na základě jejich závažnosti.
- V matrici jsou rizika obvykle rozdělena do kategorií na základě jejich pravděpodobnosti a dopadu (například vysoká pravděpodobnost, vysoký dopad; nízká pravděpodobnost, vysoký dopad atd.).
- Pomáhá organizaci identifikovat klíčová rizika, která je třeba řešit prioritně, a určit vhodné řídicí opatření
- Mez akceptovatelnosti – linie, která vymezuje akceptovatelnost určité události



Mapa rizik:

- Mapa rizik je grafické znázornění rizik a jejich vzájemných vztahů.
- Používá se k vizualizaci rizikového prostředí a k identifikaci klíčových rizikových oblastí.
- Na mapě jsou rizika zpravidla zobrazena jako uzly (nebo body) spojené čarami, které představují jejich vztahy.
- Pomáhá organizaci lépe porozumět složitosti rizikového prostředí a určit priority při řízení rizik.

Pravděpodobnost výskytu						
	velmi vysoká	5	10	15	20	25
	vysoká	4	8	12	16	20
	střední	3	6	9	12	15
	nízká	2	4	6	8	10
	velmi nízká	1	2	3	4	5
		velmi nízká	nízká	střední	vysoká	velmi vysoká
	Dopady (Škody)					

Kódování a šifrování

Kódování

- Převod informace se známým zpětným postupem
- Neslouží primárně k utajení dat, jen k jejich lepšímu zápisu (morseovka)

Šifrování

- Převod informace, je třeba znát tajný klíč
- Slouží k utajení informace
-

Symetrické šifrování

- má 1 klíč – stejný pro zašifrování i rozšifrování
- dobrá k zašifrování vlastních dat
- nevhodný pro šifrovanou komunikaci – problém, když šifra unikne
- dělení – Proudová, Bloková

Proudová

- šifrujeme postupně bit po bitu
- streamy
- RC4 – protokoly SSL, VEP, VPA
- CHACHA20

Bloková

- data rozsekáme na bloky stejné velikosti
- šifrujeme celé bloky najednou
- pokud nemají bloky stejnou velikost, tak se udělá padding – dopočtení

DES – Data Encryption Standard

- starší šifra, 64bitové bloky
- klíč – 56 bitů – zbylých 8 = parita
- dnes není doporučený – dnes se lehce prolomí
- odvádí se 16 kol šifrování

AES – Advanced Encryption Standard

- 128bitové bloky, velikost klíče – 128, 192, 256 bitů
- podle velikosti klíče se odvádí kola šifrování – 10, 12, 14
- bezpečnější, náročnější na výpočty, čas
- efektivní na HW i SW
-

3DES

- zpětně kompatibilní s DES
- pomalejší než AES
- 56bitové klíče
- zůstává v aplikacích, kde by bylo složité přejít na AES

Blowfish

- rychlá a efektivní v SW implementacích
- 64bitové bloky
- variabilní délka klíče – 32-448 bitů

Twofish

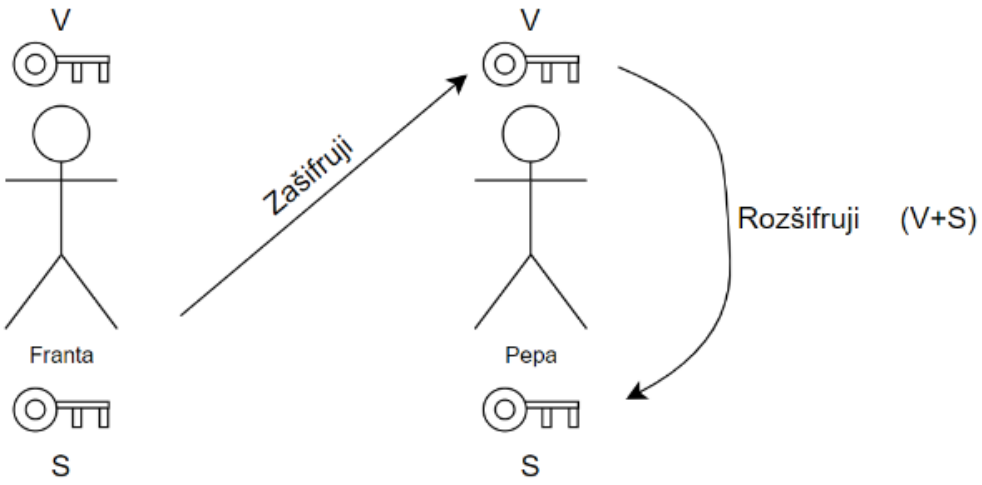
- 128bitové bloky
- klíče až do 256 bitů

Serpent

Kuznyechik

- ruský, 2015

Asymetrické šifrování



- každý má 2 klíče – Veřejný a Soukromý
- zašifrujeme cizím veřejným klíčem (Franta zašifruje Pepovo veřejným klíčem)
- Pepa rozšifruje pomocí svého veřejného a soukromého klíče (jsou matematicky propojeni)
- já už zpátky nerozšifruji, protože nemám jeho soukromý klíč
- veřejný klíč můžeme klidně dát na internet – dokud nemají soukromý klíč, tak jim je
- samotný veřejný k ničemu

RSA – Rivesta Shamir Adelman

- první algoritmus, který byl vhodný jak pro šifrování, tak pro podepisování
- nejvíce rozšířený algoritmus
- délka klíče – 2048 bitů – nejmenší velikost, běžné používání
- 3072 bitů
- 4096 bitů – nejbezpečnější – pomalejší

DSA – Digital Signature Algoritm

- ideální pro elektronické podpisy

ElGamal

- používaný v PGP – Pretty Good Privacy
- pro elektronické podpisy, v e-mailech

Hash

- matematická funkce – převádí zprávu na kód
- pro stejný vstup je vždy stejný hash

- nezáleží na velikost dat – hash bude mít stejnou velikost
- skoro nemožné získat data zpět
- malé změny ve vstupních datech vedou k velkým změnám v hash hodnotě
- využití – ukládání hesel, digitální podpisy, kontrola integrity dat, rainbow tables, …

CRC kód – Cyclic Redundancy Check

- vychází z hashe
- jednoduchý způsob kontrolního součtu
- zkontroluje hash na začátku a na konci
- používá se k detekci chyb během přenosu, či k ukládání dat

Salt

- Salt je náhodná data použitá jako další vstup pro hashovací funkci.
- Použití soli spolu s heslem zvyšuje bezpečnost hashování, protože dvě stejná hesla budou mít odlišné hash hodnoty, pokud jsou použity různé soli.

Pepper

- Pepper je další forma soli, ale na rozdíl od soli, která je ukládána spolu s heslem v databázi, je pepper typicky uložen mimo databázi.
- Je to další vrstva ochrany pro hesla v případě, že databáze je kompromitována, protože útočníkovi bude těžší získat pepper než sol.

Iterace

- Iterace se týká opakování určité operace nebo procesu, jako je hashování nebo šifrování, vícekrát.
- Použití iterací zvyšuje náročnost pro útočníka, který se snaží získat původní data z hash hodnoty nebo zašifrovaných dat. Čím více iterací, tím delší je časová náročnost útoku hrubou silou.