**Integrita dat, bezpečnost, logování, kontrola vstupů, zpracování chyb**

**Bezpečnost**

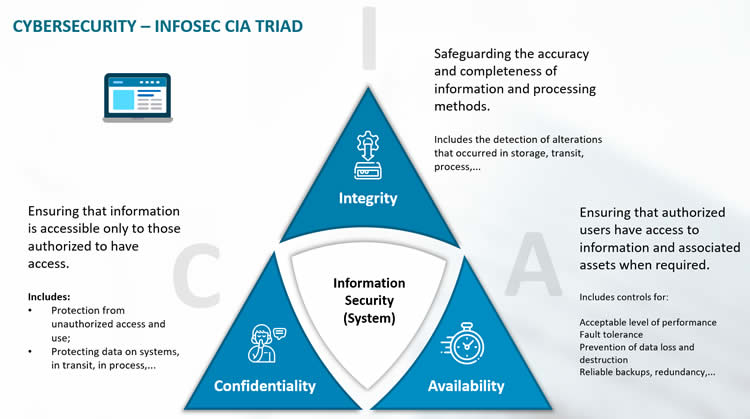
Co to vlastně je?

**Ochrana SW systémů před neautorizovaným přístupem, zneužitím, modifikací nebo degradací**

**Cílem je minimalizovat riziko útoků na aplikace a ochránit citlivá data**

**Dělí se většinou na 3 základní třídy (CYBERSECURITY CIA TRIAD)**

* Ten se používá jako rámec pro hodnocení úrovně zabezpečení informací
* Je důležité zajistit, že všechny tři prvky jsou zabezpečeny na vysoké úrovni – **CELKOVÁ BEZPEČNOST**
  1. **Confidentiality** -> Důvěrnost
  2. **Integrity** -> Integrita
  3. **Availability** -> Dostupnost
* Tři nejdůležitější koncepty zabezpečení



**Confidentiality**

* Uložení dat privátně nebo tajně
* Zabránění neoprávněnému přístupu
  + Autorizace
* Příklad útoku na tuto bezpečnost
  + „Man in the middle“ attack pro získání údajů
* Příklad chyby, která zanedbává ochranu
  + nevhodné sdílení údajů
  + nešifrované sdílení údajů
  + slabý heslo
* **Jak implementovat?**
  + Přidělovat přístupy
  + Šifrování dat (např. 2FA)
  + Školení zaměstnanců

**Integrity**

* **Zajišťuje** konzistenci, bezchybnost, úplnost, důvěryhodnost
* **Zamezuje** korupci nebo změnu dat kvůli chybě
* Příklad integrity
  + web obsahuje správná a bezchybná data
  + při přenosu dat se žádná data neztratila
* **Jak zajistit integritu?**
  + Vždy provádět validaci
  + Implementovat „Access controls“ – přístupy
  + Implementovat logování
  + Backup
  + Školení zaměstnanců

**Availability**

* Snaha o zajištění rychlého a snadného přístupu pro povolené osoby
  + Např. dostupnost kriticky potřebných služeb při výpadku
  + Např. zamezení shození serveru pomocí DoS útoků
* **Jak dostupnost zajistit?**
  + Implementování několika sítí/serverů
  + Včasná aktualizace
  + Monitorovací systémy – zaznamenání výpadků
  + Zajistit plán obnovení a pokračování v případě ztráty

**Integrita dat**

Jedna z částí bezpečnosti

**Zajišťuje** konzistenci, bezchybnost, úplnost, důvěryhodnost

**Zamezuje** korupci nebo změnu dat kvůli chybě

Zachování dat podle jejich plánu

* Např. Integrita v DATABÁZI
  + Entitní
    - Každá tabulka musí mít PK
  + Referenční
  + Doménová
    - Hodnoty ve sloupci splňují definovaný datový typ
    - Splňují uživatelsky (administrátorem) definovaný pravidla (CHECK)

**Logování**

Funkce, která zaznamenávají určitá data za účelem jejich analýzy

* Jejich důvod a cíle
  + **Chyby**
    - pro nalezení důvodu vyskytnutého problému (dohledání sekvenci provedených akci/operací kteří způsobili danou chybu)
  + **Informace**
    - Informační logy
  + **Problémy**

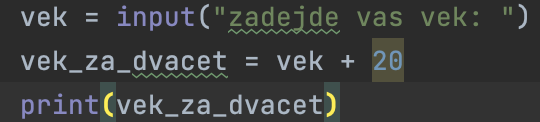
A picture containing text, screenshot, font

Description automatically generated

**Kontrola vstupů a zpracování chyb**

* Pro zachování bezpečnosti je nutno kontrolovat všechny vstupy do programu
  + **Uživatelské vstupy**
  + **Zapisované soubory**
  + **Konfigurační soubory**
* **Nejlepší způsob, jak navrhnout kontroly, je předpokládat, že uživatel je hlupák!**
  + Uživatel zadá místo "ano" např. "Ano"
  + Zadání jiného datového typu, než je očekávaný
* K ošetření vstupů např. věku od uživatele budeme kontrolovat datový typ
  + Při špatném datovém typu, nám program může spadnout při provádění následujících operací se vstupem

Požadujeme zadání věku pro výpočet zadaného věku za 20 let



* Pokud zadáme INT -> věk se spočítá správně, pokud ale zadáme STRING -> vyhodí se python chyba a program se ukončí (spadne), proto musíme vstup ošetřit
  + Vstup můžeme ošetřit například pomocí „chytání“ chyby

Text

Description automatically generated

Definování vlastní chyby

A picture containing text, font, screenshot

Description automatically generated

A picture containing text, font, screenshot

Description automatically generated