

# EVIDENCE PROCESŮ

Evidence procesů zahrnuje správu a sledování jednotlivých procesů, které operační systém spouští a řídí. Tato prezentace se zaměřuje na definici procesů, různé stavy procesů, datové struktury používané pro evidenci procesů v systémech Windows a Linux, mechanismy vzniku a ukončení procesů a přístupová oprávnění. Poskytuje přehled o rozdílech v evidenci procesů mezi různými operačními systémy, což je nezbytné pro pochopení fungování multitaskingu a přidělování systémových zdrojů.

# DEFINICE PROCESŮ A ÚLOH

## Proces vs. Program

- Program je kód uložený na paměťovém médiu; proces je instance tohoto kódu s vlastním stavem, prioritou a zdroji.

## Vlastnosti procesu

- Stav procesu, priorita, programový čítač, přidělené paměťové oblasti a identifikační číslo (PID).

## Procesy s vlákny

- Každý proces obsahuje minimálně jedno vlákno (hlavní vlákno), které může být doplněno dalšími vlákny pro paralelní běh.

## Úloha

- Ve vícevláknových systémech úloha představuje sekvenční výpočet, často realizovaný jedním vláknem.

# STAVY PROCESU

## Nový (new)

- Proces byl vytvořen a jsou mu přidělovány systémové prostředky.

## Běžící (running)

- Proces má procesor a aktivně vykonává kód.

## Připravený (ready)

- Proces čeká na přidělení procesoru.

## Čekající/Blokovaný (waiting/blocked)

- Proces čeká na vstupně-výstupní operaci nebo jinou událost.

## Ukončený (terminated)

- Proces skončil, ale systémové struktury ještě existují.

# PŘECHODY MEZI STAVY PROCESU

## Přidělení a odebrání procesoru

- Přechody mezi stavy „připravený“ a „běžící“.

## Čekání na I/O zařízení

- Přechod z „běžícího“ do „čekajícího“.

## Dokončení události

- Přechod z „čekajícího“ do „připraveného“.

## Schéma stavů procesu

- Grafická ilustrace přechodů mezi jednotlivými stavy.

# BINÁRNÍ SPUSTITELNÉ SOUBORY

## Formát spustitelných souborů

- Ve Windows: formáty PE, PE32 a další; v Linuxu: ELF, a.out.

## Statické vs. Dynamické linkování

- Staticky linkované programy obsahují veškerý kód, dynamicky linkované volají knihovny za běhu.

## Rozpoznání formátu

- První bajty souboru často indikují formát (např. ELF pro Linux, PE pro Windows).

# DATOVÉ STRUKTURY PRO EVIDENCI PROCESŮ

## Process Control Block (PCB)

- Obsahuje informace potřebné pro řízení procesu (PID, stav, registr, fronty, paměťové tabulky).

## Struktury ve Windows

- **EPROCESS** (pro procesy), **ETHREAD** (pro vlákna), **PEB** a **TEB** pro prostředí procesu a vláken.

## Struktury v Linuxu

- **task\_struct** v jádře Linuxu, obsahující stav procesu, informace o rodiči, potomcích, přidělených zdrojích.

## Redundantní data

- Ve víceúrovňových strukturách dochází k určité redundanci pro optimalizaci přístupu.

# PRIORITY PROCESŮ

## Základní, dynamická a statická priorita

- Základní: přiřazena při vzniku, neměnná. Dynamická: dočasně mění prioritu. Statická: pro reálné procesy.

## Priority ve Windows

- Rozsah 1–31; hodnoty 1–15 pro běžné procesy, 16–31 pro procesy reálného času.

## Priority v Linuxu

- Rozsah 1–40 pro běžné procesy, 1–99 pro reálné procesy; dynamické priority jsou řízeny hodnotou „nice“.

# MECHANISMY VZNIKU A ZÁNÍKU PROCESŮ

## Vznik procesu

- Spuštění programu rodičem, nebo klonováním procesu pomocí `fork ()` (Linux) nebo `CreateProcess ()` (Windows).

## Stromová struktura procesů

- Ve víceprocesorových systémech tvoří procesy stromovou strukturu s „rodičem“ a „potomkem“.

## Zánik procesu

- Proces se ukončí buď sám (např. pomocí `exit ()`), nebo je ukončen rodičem (v Linuxu signálem, ve Windows metodou `abort ()`).

## Možnosti ukončení

- Ukončení sebe, potomků, sirotků nebo násilné ukončení uživatelem.



# PŘÍSTUPOVÁ OPRAVNĚNÍ PROCESU

## Dědění oprávnění

- Procesy obvykle dědí oprávnění od rodičovského procesu.

## Získání oprávnění obrazu

- Speciální práva procesu mohou být nastavena na základě vlastností spustitelného souboru.

## Změna oprávnění během běhu

- Pomocí mechanismů jako „**Run As**“ ve Windows nebo **sudo** v Linuxu je možné změnit oprávnění procesu.

## Bezpečnostní opatření

- V Linuxu lze zvýšit oprávnění pouze superuživatelé, čímž je zajištěna stabilita a bezpečnost systému.

# SHRNUTÍ

- Proces je definován svým stavem, zdroji a identifikačním číslem, což ho odlišuje od programu.
- Stavy procesu (nový, běžící, připravený, čekající, ukončený) ilustrují různé fáze životního cyklu procesu.
- Různé OS mají specifické datové struktury (např. PCB, **task\_struct**) pro správu procesů a zdrojů.
- Priority určují, kdy proces získá přístup k CPU; dynamické priority se mění dle zátěže systému.
- Přístupová oprávnění jsou klíčová pro bezpečnost a řízení, umožňují řízené sdílení zdrojů.

# KONTROLNÍ OTÁZKY

1. Jaký je rozdíl mezi procesem a programem?
2. Jaké jsou základní stavy procesu a jaké události mezi nimi vyvolávají přechody?
3. Co je PCB a jaké informace v něm OS uchovává?
4. Jakým způsobem mohou být v Linuxu a Windows změněna přístupová oprávnění procesu?
5. Co znamená dynamická priorita procesu a jak je nastavována?

# DOPORUČENÁ LITERATURA

1. **Silberschatz, A., Galvin, P. B., & Gagne, G.** - *Operating System Concepts* - Klíčová témata správy procesů.
2. **Tanenbaum, A. S., & Bos, H.** - *Modern Operating Systems* - Detailní popis mechanismů procesů v OS.
3. **Stallings, W.** - *Operating Systems: Internals and Design Principles* - Detailní struktura procesů.
4. **Love, R.** - *Linux Kernel Development* - Správa procesů a priorit v Linuxovém jádře.
5. **Microsoft Documentation on Process Management** - Oficiální dokumentace k evidenci procesů ve Windows.