

strace

monitoruje systémová volání a signály pomocí speciálního syscallu ptrace, který slouží pro trasování běžícího kódu programu, ptrace je rozsahlá a velmi silná záležitost, dokáže se napojit na proces, krokovat ho, zasahovat do jeho paměti, odchytávat signály apod, je také využíváno oblíbeným GNU debuggerem gdb

komunikace procesů v FreeBSD

- Systémová volání
přes VOLÁNÍ FUNKCE, která obsahuje int 0x80
nebo přes volání brány call 7:0
- Číslo systémového volání
v registru eax
- Parametry systémového volání
na zásobníku (první parametr je ukládán poslední)
- Výsledek
v registru eax
volající proces musí vyčistit zásobník (parametry)

přerušování(Round Robin) způsobí

proces přejde ze stavu running do stavu ready

synchronní I/O způsobí

proces přejde ze stavu running do stavu waiting

paralelní vs. distribuovaný systém

rozdíl ve sdílení/nesdílení paměti

IPC

semafor

sdílená paměť

roury

Předávání argumentů (parametrů) při systémových volání je realizované

registry

zásobník

pointer na strukturu s daty uloženými v paměti patřícího procesu (blok v paměti & pointeru)

pthread_create()

vytvoření vlákna

POSIX

CreateThread()

vytvoření vlákna

Win32

dlouhodobý (strategický) plánovač

vybírá který proces lze zařadit mezi připravené procesy

plánovač je spouštěn je relativně málo často – typicky při ukončení jednoho procesu rozhodne, kterou úlohu dále vybrat k zavedení do paměti a spuštění

nemusí být super rychlý

určuje stupeň multiprogramování

střednědobý (taktický) plánovač

vybírá který proces lze zařadit mezi odložené procesy

vybírá který odložený proces lze zařadit mezi připravené procesy

krátkodobý plánovač

Přiděluje procesor připraveným procesům

Je spouštěn často (např. každých 10ms)

Proto musí být rychlý

RES

rezidentní paměť, která nebyla od swapovaná

(rezidentní čas, neodswapovaná fyzická paměť, který proces použil)

fork()

vytvoří nový proces v Linuxu

jeho ekvivalent ve WIN API je createProcess()

OS s mikrojádrem

je stabilnější a bezpečnější

snadná přenositelnost OS

vlákna

sdílejí stejný adresový prostor, číslo procesu, číslo vlákna

plánovací algoritmy na Linuxu

Noop

Anticipatory

Deadline

CFQ (Completely Fair Queue)

nutná podmínka uvážnutí

- vzájemné vyloučení (mutual exclusion)
sdílený zdroj může v jednom okamžiku používat pouze jeden proces
- ponechání si zdroje a čekání na další (hold and wait)
proces vlastní alespoň zdroj čeká na získání dalšího zdroje, dosud vlastněného jiným procesem
- bez předbírání (no preemption)
zdroj lze uvolnit pouze procesem, který ho vlastní, dobrovolně po té, co daný proces zdroj dále nepotřebuje
- kruhové čekání (circular wait)
existuje takový seznam čekajících procesů (P_0, P_1, \dots, P_n), že P_0 čeká na uvolnění zdroje drženího P_1 , P_1 čeká na uvolnění zdroje drženího P_2, \dots, P_{n-1} čeká na uvolnění zdroje drženího P_n , a P_n čeká na uvolnění zdroje drženího P_0

multitasking

snižuje dobu odezvy

sys. volani MS-DOS

int 0x21

Linux

modulární monolitické jádro