

Nume și grupă:

Sisteme de Operare

30 august 2018

Timp de lucru: 90 de minute

Notă: Toate răspunsurile trebuie justificate

- (7 puncte)** Un sistem de operare are 10 procese în starea **RUNNING**. Câte procesoare are sistemul fizic?
- (7 puncte)** Scrieți o secvență de cod care permite unui proces să scrie direct în memoria unui alt proces de pe același sistem.
- (7 puncte)** Cu cât va crește consumul de memorie al unui proces atunci când acesta creează un thread nou?
- (7 puncte)** Procesul 1 scrie în octetul de la adresa virtuală **0x400100** valoarea **111**. Imediat după, procesul 2 citește de la aceeași adresa virtuală (**0x400100**) valoarea **222**. Cum se explică acest lucru?
- (7 puncte)** Explicați ce se întâmplă atunci când un proces încearcă să acceseze o adresă virtuală nevalidă.
- (10 puncte)** Un planificator de procese folosește o cuantă de 8 microsecunde. Contextul unui proces conține 10 valori de 32 biți, iar timpul de access de memorie (*read* sau *write*) pentru un cuvânt de 32 biți este de 100 nanosecunde. Care va fi overhead-ul de planificare pe acest sistem, dacă algoritmul de planificare rulează instantaneu?
- (10 puncte)** Un server care folosește `fork() + exec()` pentru fiecare conexiune este atacat printr-un atac de tip *buffer overflow*. Serverul folosește *stack canaries* pe 64 biți. Cum poate fi serverul exploatat?
- (10 puncte)** Un proces are deschis un fișier. Procesul își încheie execuția. Cu toate acestea, structura aferentă fișierului deschis nu este eliberată. Cum explicați?
- (10 puncte)** După un timp de rulare a codului de server de mai jos, serverul nu mai funcționează coresponszător. Care este problema, presupunând că toate apelurile de sistem se întorc cu succes?

```
while (1) {
    int fd = accept(sock);
    if (fork() == 0) {
        read(fd, buf, 100); write(fd, msg, 100);
        close(fd);
        exit(0);
    }
}
```

- (10 puncte)** Un utilizator a rulat din greșeala comanda `rm fisier_favorit.txt` pe un sistem de fișiere ext2 (sistem de fișiere fără jurnalizare). Ce poate face pentru a încerca să recupereze datele din fișier? Nu ne referim la un răspuns *folosim un utilitar de pe Internet*.

11. (25 de puncte) Un proces părinte trebuie să se sincronizeze cu procesul copil cu ajutorul semafoarelor, însă sistemul de operare nu oferă suport de semafoare. Sistemul de operare oferă însă suport pentru memorie partajată (via `mmap()`) și pentru pipes. Furnizați cod C care implementează:

- a) o structură de date unde vor fi salvate informațiile pentru semafor **(5 puncte)**
- b) apelul `int sem_create(int initial_val)` care întoarce un întreg reprezentând semaforul dorit și îl inițializează la `initial_val` **5 puncte**
- c) funcția `sem_put(int sem_id)` care incrementează semaforul **(5 puncte)**
- d) funcția `sem_get(int sem_id)` care decrementează semaforul dacă acesta nu este 0; altfel se blochează până când semaforul este incrementat, și apoi realizează decrementare și se întoarce **(5 puncte)**
- e) cod care folosește funcțiile de mai sus **(5 puncte)**

În conformitate cu ghidul de etică al Departamentului de Calculatoare, declar că nu am copiat și nu voi copia la această lucrare. De asemenea, nu am ajutat și nu voi ajuta pe nimeni să copieze la această lucrare.

Nume și grupă:

Semnătură:.....