**Modul Kernel pentru inspectie structuri de date**

**Grupa: C-113 C**

**Studenți:**

**Sd. Sg. Huțanu David**

**Sd. Sg. Niță Claudiu**

**Scopul proiectului:**

Acest modul de kernel se adresează utilizatorilor care doresc sa verifice funcționalitatea anumitor structuri de date (linked list, red black tree, bitmap, etc.) sau sa afle informatii despre sistem stocate in structuri de date mentionate mai sus (process table, I/O queues, descriptor table etc.).

Pentru usurarea utilizatorului, in cadrul modulului a fost implementata o functionalitate DeviceIOControl (IOCTL) care permite transmiterea de parametri catre modul fara a mai fi nevoie de a descarca si incarca manual modulul cu noii parametri.

In acest sens a fost implementat in fisierul /usr/bin un script care verifica daca modulul ce se vrea a fi incarcat are parametri, daca modulul este deja incarcat, daca nu este il incarca, si care creeaza un fisier /dev/kds prin care se face comunicarea dintre userspace si modul.

Dupa ce sunt indeplinite conditiile de mai sus, scriptul pune in executie un program „test.c” care va prelua parametrii transmisi de utilizator si ii va scrie in fisierul /dev/kds.

Modulul, dupa incarcare, supravegheaza fisierul /dev/kds iar cand acesta este modificat, citeste valoarea care este scrisa in fisier si o parseaza, executand anumite operatii in functie de parametrii transmisi.

Astfel, este suficient ca utilizatorul sa scrie comanda „*sudo kds (-lista de parametri)*” in terminal pentru a executa o rulare a modulului. Parametrii disponibili sunt: -b/--basic; -p/--process; -f/--file; -d/--device. In functie de acestia, pot realizate urmatoarele operatii:

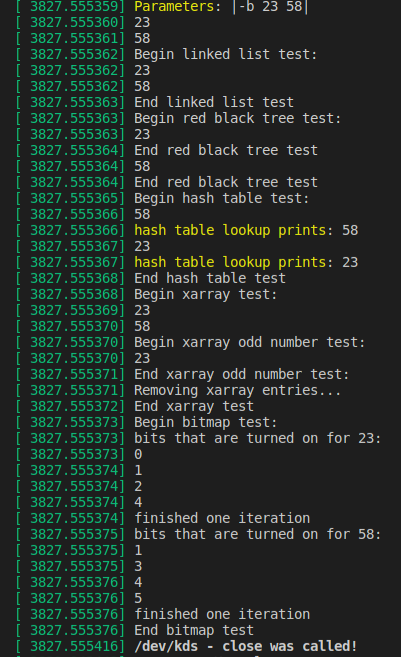
1. **Verificarea structurilor de date**

La primirea parametrului -b/--basic, urmat de o liste de numere intregi, modulul va efectua verificari ale structurilor de baza din kernelul de linux folosind numerele date ca parametru. Structurile care vor fi verificate sunt:

* Linked list
* Red black tree
* Hash table
* Radix tree
* Xarray
* Bitmap

Astfel, pentru fiecare numar care va fi transmis, va fi alocata o astfel de stuctura, populata cu respectivul numar, afisat rezultatul, apoi dezalocata.

De exemplu, daca introducem numerele 23 si 58, va fi afisat rezultatul:



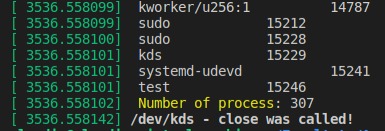
1. **Afisarea proceselor existente si informatii despre acestea**

La primirea parametrului -p/--process, modulul va returna

informatii despre procesele care ruleaza in respectivul moment pe masina. Aceste informatii difera in functie de numarul de parametri dat de catre utilizator.

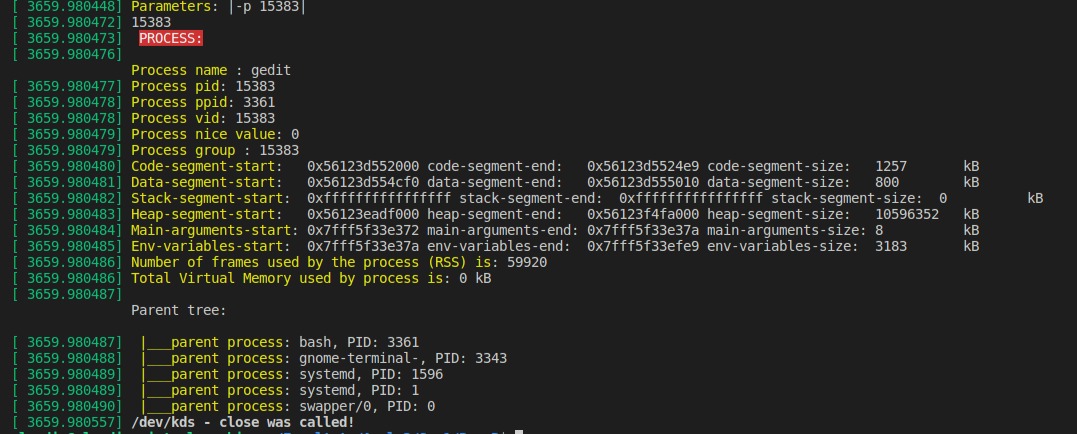
Daca se va introduce doar parametrul -p/--process, modulul va interoga structurile existente si va returna pidul si numele tuturor proceselor existente in acel moment tinand in acelasi timp o evidenta a numarului de procese, pe care il va afisa la sfarsit.

Astfel, la executarea comenzilor „sudo kds -p” urmata de „sudo dmesg”, utilizatorul va vedea un rezultat asemanator cu cel de mai jos:



Pe de alta parte, daca pe langa parametrul -p sa va da si o lista de unul sau mai multe pid-uri existente in sistem, modulul va afisa informatii specifice despre fiecare proces specificat in parte. Aceste informatii se refera la ...

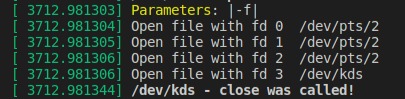
Un exemplu de rezultat care poate fi intors poate fi:



1. **Afisarea descriptorilor existenti si informatii despre acestia**

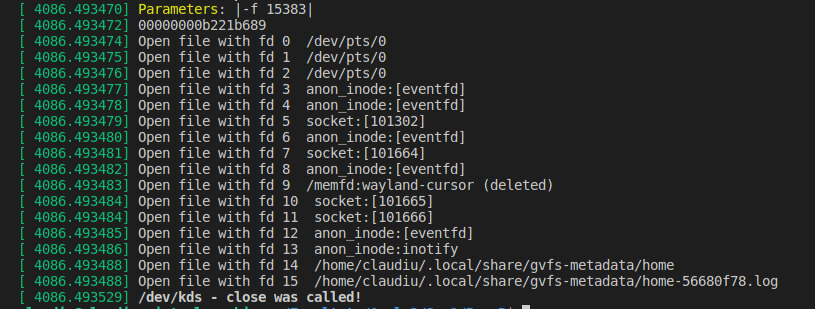
La primirea parametrului -f/--file, modulul va returna informatii despre descriptorii de fisiere care sunt deschisi la un moment de timp in sistem. De asemenea, in functie de tipul si numarul parametrilor transmisi, modulul va returna o serie de informatii.

Daca se va introduce doar parametrul -f/--file, modulul va intoarce toti descriptorii si numele de fisier deschise de procesul curent. Astfel, dupa executarea comenzii „sudo kds -f” in terminal, se va intoarce un rezultat asemanator cu acesta:



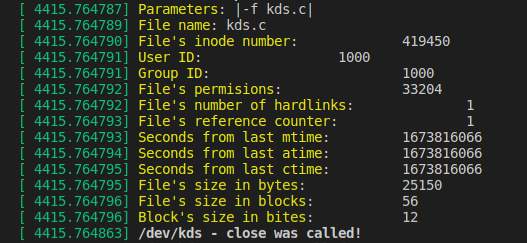
Pe de alta parte, daca pe langa parametrul -f sa va da si o lista de unul sau mai multe pid-uri existente in sistem, modulul va afisa informatii specifice despre fiecare descriptor deschis de procesele ale caror pid-uri au fost transmise. Aceste informatii se refera la file descriptorului fisierului precum si numele acestuia.

Un exemplu de rezultat care poate fi intors poate fi:

****

De asemenea, daca pe langa paramentrul -f se va da o lista de unul sau mai multe nume de fisiere deschise in acel moment, modulul va intoarce informatii specifice despre respectivele fisiere: inode, permisiuni, timpi de acces, marimea fisierului, etc.

Un exemplu de rezultat care poate fi intors poate fi:



1. **Afisarea cozilor de I/O existente precum si informatii despre fiecare element din acestea**

La primirea parametrului -d/--device, am incercat sa implementam functionalitatea de afisarea a elemntelor din structura request\_queue.

De asemenea pentru facilitatea utilizatorului si din consideratii de securitate, cu ajutorul serviciului crontab din linux acest modul se executa periodic la 6 ore cu parametrii -p si -f afisand detaliile corespunzatoare in dmesg.