  @@@

@@@

@@@

@@@

@@@@@@@@@@@

@@ .@@

@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@

@@ @@

@@ @@@@@@@@ @@@@@@@ @@@@@@@& @@@@@@@ @@@@@@@/ @@@@@@@@@@@@@@@@@. @@

@@ @@@ @@@ @@ @@ @@@ @@& @@ \*@@ @@@ @@/ @@ @@. @@

@@ @@@@@@@@ @@@@@@@ @@@@@@@& @@@@@@@ @@@@@@@/ @@@@@@@@@@@@@@@@@. @@

@@ @@

@@ @@@@@@@@ @@@@@@@ @@@@@@@& @@@@@@@ @@@@@@@/ @@@@@@@ @@@@@@@. @@

@@ @@@ @@@ @@ @@ @@@ @@& @@ \*@@ @@@ @@/ @@ %@@ @@@ @@. @@

@@ @@@@@@@@ @@@@@@@ @@@@@@@& @@@@@@@ @@@@@@@/ @@@@@@@ @@@ @@. @@

@@ @@@ @@. @@

@@ @@@@@@@@ @@@@@@@@@@@@@@@@@@@@@@@@ @@@@@@@ @@@ @@. @@

@@ @@@ @@@ @@ %@@@@@@@@@@@@@@@ @@ %@@ @@@ @@. @@

@@ @@@@@@@@ @@@@@@@@@@@@@@@@@ @@@ .@ @@@@ @@@@@@@ @@@@@@@. @@

@@ @@@ @@@@@ (@@, @@

@@@@@@@@@@@@@@@@@@@@@@@@@@@@@ @@@ @@@ @@ @@@@@@@@@@@@@@@@@@@@

\*@@ @@@ @@@

\*@@ @@ \*@@

@@/ @@& @@&

@@@ @@@@ @@@@@@

@@@@ @@@ @@@ @@@@

@@@@@@ @@@@@&(@@@@ @@@@@

%@@@@@ @@@@, @@@@

\*@@@@ @@@@@

@@@@( @@@@

@@@@ @@@(

@@@@@@

**KEYLOGGER-D.N.L**

“Proiectarea Sistemelor de Operare – Proiect”

Proiect realizat de:

* Sd. Sg. Dumitrascu Andrei-Cosmin
* Sd.Sg Mitran Luca-Radu

ANUL III

***Cuprins:***

1. Scopul proiectului …………………………………………….. pagina 3
2. Lista definitiilor …………………………………………………. pagina 3
3. Descrierea produsului software ………………………… pagina 4
4. Detalierea platformei HW/SW ………………………….. pagina 5
5. Cerinte functionale …………………………………………… pagina 5

**Scopul proiectului:**

Scopul proiectului realizat de echipa noastra este de a implementa un keylogger ce inregistreaza toate apasarile de taste intr-un modul de kernel de linux.

Se vor implementa, de asemenea si algoritmi de detectie si predictie pentru parole ale adreselor de email sau conturi bancare.

**Lista definitiilor:**

1. **IRQ:** interrupt request handle
2. **ISR:** interrupt service routine
3. **Tasklet**: Tasklet-urile sunt folosite pentru a pune în coadă job-urile care urmează să fie efectuate ulterior

**Descrierea produsului software:**

Proiectul descrie si implementeaza un keylogger pe un modul de kernel ce va stoca pe un fisier .log toate intrarile de la tastatura.

Pentru ca toate datele inregistrate sa fie stocate intr-un mod organizat se vor salva toate datele in fisiere denumite cu numele utilizatorului.log si cu ajutorul algoritmilor implementati de noi se vor gasi parolele conturilor de email si ale conturilor bancare sau ale diverselor metode de logare existente.

Implementarea in faza de testare se va face in 2 moduri:

1. Fara utilizarea modulului de kernel: se citesc tastele apasate dintr-un fisier /dev/input/event
2. Prin utilizarea modulului de kernel: se asteapta intreruperile de tasta si apeleaza functiile necesare.

**Detalierea platformei HW/SW**

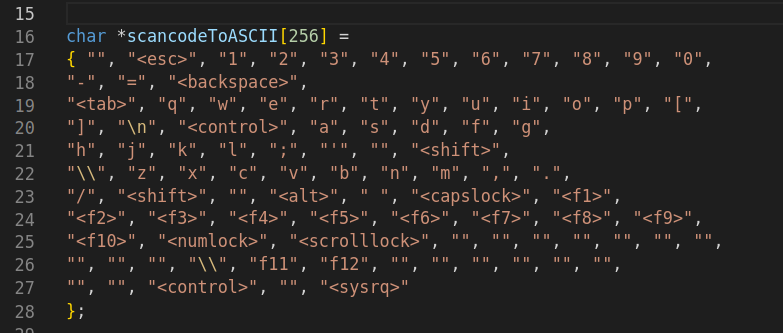
Keyloggerul va rula pe sistemul de operare Linux si va fi dezvoltat pe 20.04LTS

**Cerinte functionale**

Keylogger-ul se va folosi de functia int request\_irq(unsigned int irq,void (\*handler)(int, void \*, struct pt\_regs \*),unsigned long flags, const char \*dev\_name, void \*dev\_id); pentru a accesa intreruperile folosite de driver-ul de tastatura la apasarea unei taste. Vom folosi functia pentru a lega un handler de intrerupere pentru intreruperile efectuate la apasarea unei taste.

Handler-ul irq\_handler\_t se va folosi de un tasklet pentru a optimiza accesul la resursele procesorului, mai exact pentru scrierea buffer-ului cu codurile ASCII captate intr-un fisier de log.

Se va capta scancode-ul trimis de tastatura si acesta va fi translatat in ASCII folosind un vector indexat de scancode-uri cu elemente char\*:



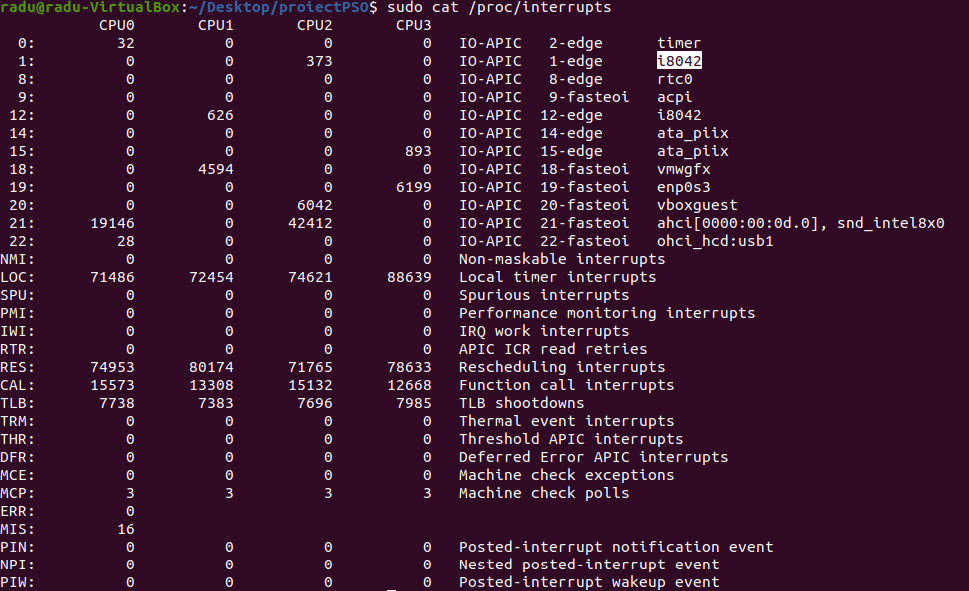
In cazul folosirii <caps\_lock> si apoi o secventa de taste keylogger-ul va afisa automat secventa UPPERCASE. La urmatorul <caps\_lock> se trece in lowercase.

test<lowecase><caps\_lock>TEST<UPPERCASE><caps\_lock>test<lowercase>

De asemenea, <shift> in combinatie cu [0-9] va jurnaliza in dreptul numarului <[!@#$%^&\*()]>. Astfel putem identifica spre exemplu posibile adrese de email:

luca.mitran<shift>2<@>mta.ro

Intreruperile sunt jurnalizate in /proc/interrupts si in /proc/stat in Linux. Folosim fisierul pentru a afla codul de intrerupere folosit de “keyboard” astfel incat keyloggerul sa fie portabil si sa autodetecteze IRQ number-ul care trebuie folosit.

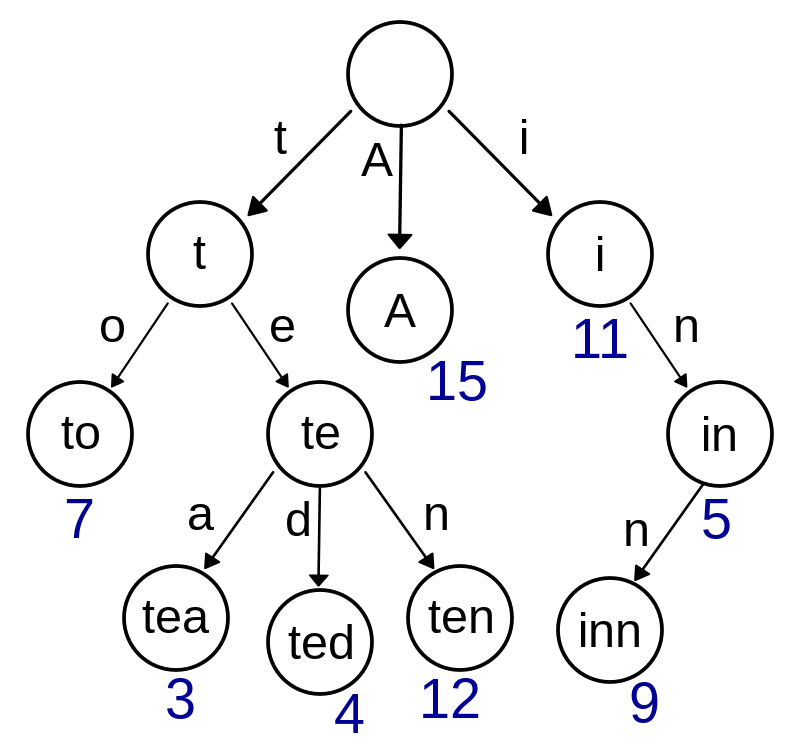
Un snapshot al /proc/interrupts:

In exemplu IRQ number-ul folosit la intreruperile de tastatura este 1 (prima coloana), iar driver-ul de tastatura este i8042. Astfel se va folosi o lista de nume de drivere usual folosite. Se va face match pe numele acestora in acest fisier pentru a obtine IRQ number.

Scancode-urile convertite in ASCII vor fi puse intr-un buffer de dimensiune fixa. Cand se ajunge la BUFF\_SIZE (#define BUFF\_SIZE 100) se va jurnaliza intr-un fisier logger.txt ce s-a scris. Aici se va folosi tasklet-ul pentru eficientizarea accesului la procesor. In fisier se va scrie folosind apelul write().

Fisierul daca exista se va deschide in modul append si se va initializa cu Data si ora curenta. Altfel acesta o sa fie creat. La inchiderea fisierului se efectueaza un flush() din buffer.

Pentru initializarea modulului de kernel si incheierea acestuia se folosesc: module\_init() si module\_exit(). In module\_init() se va deschide fisierul si initializa handler-ul de intreruperi pentru tastatura, in module\_exit se va elibera handler-ul creat si se va inchide fisierul, jurnalizandu-se ora si data la care acesta a fost inchis. In caz de eroare se va logga un mesaj corespunzator.

 Pentru datele interceptate in fisier vom folosi un utilitar in care se va incarca un dictionar de cuvinte. Acesta va implementa o structura de date de tipul trie tree pentru a eficientiza cautarea cuvintelor des folosite in fisier regasite si in dictionar. Acesta este util pentru recunoasterea accesari unor site-uri frecvent folosite.

Acest utilitar va folosi si expresii regulare pentru a recunoaste posibile match-uri ale unor secvente precum introducerea unei adrese de email sau folosirea unor detalii ale unor detalii de card bancar. Toate match-urile gasite vor fi afisate in terminal pe criterii: site found, possible login data, possible card data, email address found.