

Universidad de San Carlos de Guatemala
Facultad de Ingeniería
Escuela de Ciencias y Sistemas
Sistemas Operativos 1
Ing. Sergio Arnoldo Mendez Aguilar
Aux Bryan Otoniel Ordoñez Morales
Vacaciones de Jun. 2020



USAC
TRICENTENARIA
Universidad de San Carlos de Guatemala

MODULOS KERNEL, PLANIFICACIÓN DE PROCESOS Y MONITOREO DE RECURSOS

Nombre:

Jeannira del Rosario Sic Menéndez
Fernando Vidal Ruiz Piox

Carnet:

201602434
201503984

Descripción

Se trata de desarrollar una aplicación que permita monitorear y gestionar los procesos de un servidor Linux, por medio de una interfaz web de fácil acceso desde el navegador de una computadora o de un dispositivo móvil como teléfono o Tablet.

La información de los procesos, uso de memoria RAM y uso de CPU será extraída mediante la implementación de módulos kernel que estarán adquiriendo y exponiendo esta información utilizando archivos ubicados en /proc.

Directorio /proc/

Grupos comunes de información referente al kernel agrupado en directorios y subdirectorios en /proc/.

```
> cd /proc
```

cada carpeta con número es un proceso

```

feruiz@feruiz-Aspire-E5-473:/proc
Archivo Editar Ver Buscar Terminal Ayuda
feruiz@feruiz-Aspire-E5-473:/proc$ ls
1      13      1540    1982      2133      2321      2888      3019      3112      3371      38      5487      639      7060      7411      buddyinfo      kcore      schedstat
10     1313    155     2      2135     24      2893      3023      3116      3372      3854      5582      6410      7085      7475      bus            keys       scsi
1022   136    156     2003    2136     242      2895      3047      3142      3375      39      5621      6449      7122      7487      cgroups        key-users  self
1024   137    157     2017    2138     243      29      3052      3160      3395      392      5741      6462      7181      7506      cmdline        kmsg       slabinfo
1035   138    16     2022    2148     244      2913      3056      3167      34      3940      585      6504      7182      7508      consoles       kpagecgout softirqs
1044   139    160     2024    2150     245      2920      3064      3188      3402     4      586      6518      7193      7509      cpufreq         kpagecount stat
1047   14     1603    2028    2151     246      2925      3066      3191      3487     40      588      6536      7199      7520      crypto          kpageflags swaps
1060   140    1604     2029    2152     27      2937      3067      3204     35      41      590      6539      7216      7532      devices         loadavg    sys
1065   141    1660     2079    2154      2754      2942      3070      3221      3500      417      594      657      7224      7533      diskstats       locks      sysrq-trigger
1088   142    1662     2081    2159      2757      2954      3073      3228      3506     42      602      660      7271      7587      dma             ndstat     sysvipc
1097   1428    1664     2086    2161      2758      2958      3077      3238      3512     427      6034      661      7285      7601      driver          meminfo    thread-self
1107   143    17     2088    2162      2761      2960      3083      3254      3516      4288      6044      6629      7299      7619      execdomains     nisc       timer_list
1111   1439    171     2091    2171      2775      2964      3087      3285     354      429      613      6669      7307     9      fb             modules    tty
1113   144     1742     21      2195      2777      2968      3088      33      355      430      6174     671      7308     911      filesystems     mounts     uptime
1115   1452    1760     2118     22      2786      2978      3095      333     36     432      6285      673      7313     912      fs             ntr        version
1118   1479    18      2120     2236      2789     3      3100      3355      3632      4341     629      675      7378     977      interrupts     net         uptime_signature
1121   15     1859     2121     23      28      30      3101      3362      3634      4349     632      677      7379     989      ionen          pagetypeinfo vmallocInfo
1122   150     1860     2127     2301      285      3006      3105      3363     37      4475     633      6777      7406     994      ioports        partitions vmstat
1130   151     190     2129     2304      286      3010      3108      3666      3720     4492     635      6782      7407      acpi           irq         pressure
12     152     1942     2132     2313      2885      3015      3109      3369      378      4533     637      679      7408      asound         kallsyms   sched_debug
feruiz@feruiz-Aspire-E5-473:/proc$

```

```
> cat meminfo
```

muestra información de la memoria Ram en ese instante

```
feruiz@feruiz-Aspire-E5-473:/proc
Archivo Editar Ver Buscar Terminal Ayuda
feruiz@feruiz-Aspire-E5-473:/proc$ cat meminfo
MemTotal:        3948216 kB
MemFree:         141180 kB
MemAvailable:    141224 kB
Buffers:         6396 kB
Cached:         782280 kB
SwapCached:      8532 kB
Active:          2766344 kB
Inactive:        633412 kB
Active(anon):    2607196 kB
Inactive(anon):  509988 kB
Active(file):    99148 kB
Inactive(file):  63424 kB
Unevictable:    38912 kB
Mlocked:        48 kB
SwapTotal:      487420 kB
SwapFree:       2352 kB
Dirty:          768 kB
Writeback:       0 kB
AnonPages:      2641552 kB
```

> cat cpuinfo
muestra información de los procesadores

```
feruiz@feruiz-Aspire-E5-473: /proc$ cat cpuinfo
0
GenuineIntel
6
61
Intel(R) Core(TM) i3-5005U CPU @ 2.00GHz
4
0x2e
802.678
3072 KB
0
4
0
2
0
0
yes
yes
20
yes
fpv vme de pse tsc msr pae mce cx8 apic sep ntrr pge mca cmov pat pse36 clflush dts acpi mmx fxsr sse sse2 ss ht tm pbe syscall
cp ln constant tsc arch_perfmon pebs bts rep_good nopl xtopology nonstop tsc cpuid aperfmperf pni pclmulqdq dtes64 monitor ds_cpl
s3 sdbg fma cx16 xtrr pdcm pcd sse4_1 sse4_2 xzavic movbe popcnt tsc_deadline_timer aes xsave avx f16c rdrand lahr ln abm 3dnowp
ult epb invpcid single pti ssbd ibrs ibpb stibp tpr_shadow vnmi flexpriority ept vpid ept_ad fsgsbase tsc_adjust bml1 avx2 smep b
rdseed adx snap intel_pt xsaveopt dtherm arat pln pts md_clear flush_lid
cpu_meltdown spectre_v1 spectre_v2 spec_store_bypass l1tf mds swapgs ltlb_multihit
3999.80
64
39 bits physical, 48 bits virtual
1
GenuineIntel
6
```

Primera Parte del Proyecto

Módulo de memoria (sysinfo)

crear dos archivos

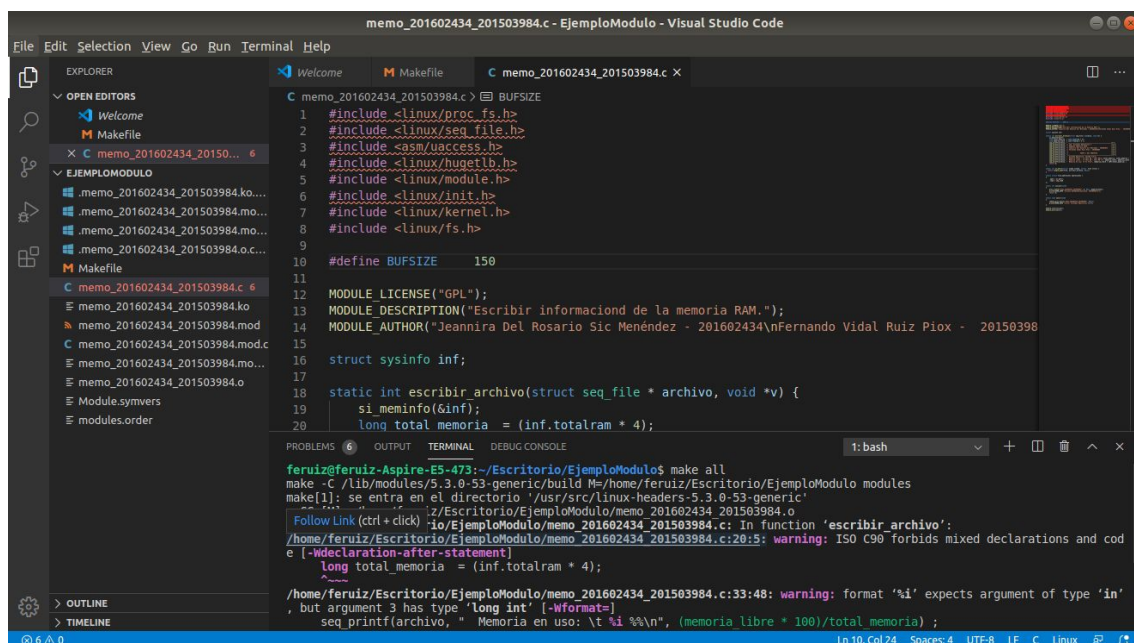
Makefile

memo_201602434_201503984.c

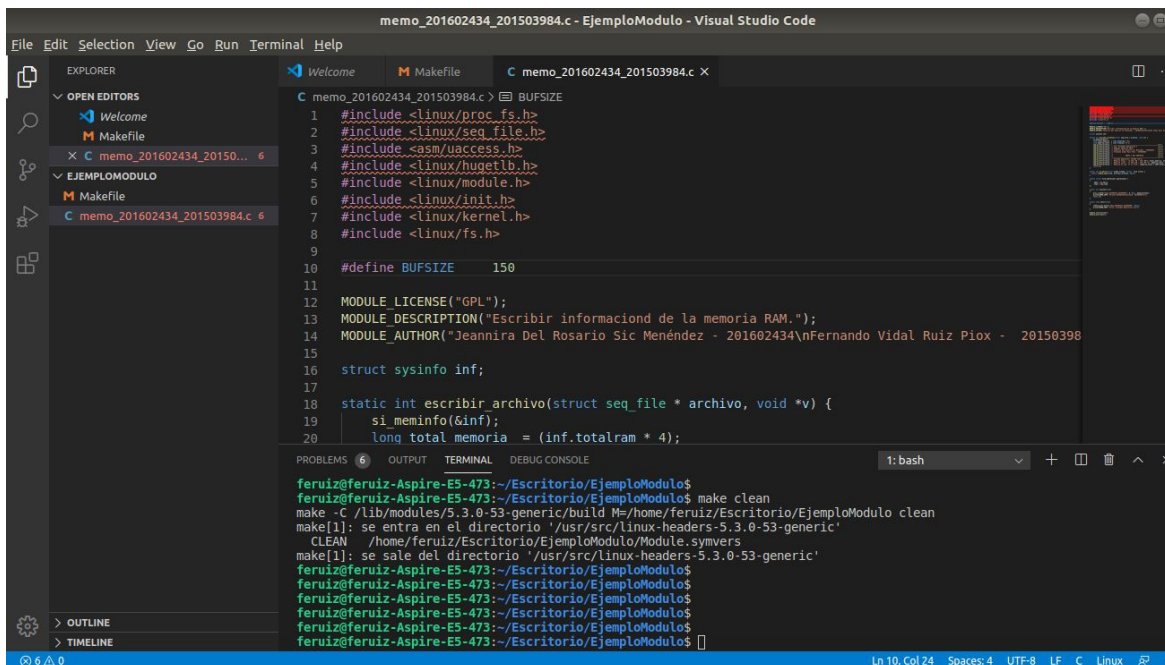
Luego se utilizan los siguiente comandos:

Para crear los archivos necesarios. (el que me importa es el archivo .ko)

> make all



Para limpiar los archivos que se crearon al ejecutar el comando anterior
> make clean



The screenshot shows the Visual Studio Code editor with the file 'memo_201602434_201503984.c' open. The code is a C kernel module. The terminal at the bottom shows the execution of 'make clean', which successfully removes the module files.

```
1 #include <linux/proc_fs.h>
2 #include <linux/seq_file.h>
3 #include <asm/uaccess.h>
4 #include <linux/hugetlb.h>
5 #include <linux/module.h>
6 #include <linux/init.h>
7 #include <linux/kernel.h>
8 #include <linux/fs.h>
9
10 #define BUFSIZE 150
11
12 MODULE_LICENSE("GPL");
13 MODULE_DESCRIPTION("Escribir informacion de la memoria RAM.");
14 MODULE_AUTHOR("Jeannira Del Rosario Sic Menéndez - 201602434\nFernando Vidal Ruiz Piox - 20150398");
15
16 struct sysinfo inf;
17
18 static int escribir_archivo(struct seq_file * archivo, void *v) {
19     si_meminfo(&inf);
20     longq total memoria = (inf.totalram * 4);
```

```
feruiz@feruiz-Aspire-E5-473:~/Escritorio/EjemploModulo$
feruiz@feruiz-Aspire-E5-473:~/Escritorio/EjemploModulo$ make clean
make -C /lib/modules/5.3.0-53-generic/build M=/home/feruiz/Escritorio/EjemploModulo clean
make[1]: se entra en el directorio '/usr/src/linux-headers-5.3.0-53-generic'
CLEAN /home/feruiz/Escritorio/EjemploModulo/Module.symvers
make[1]: se sale del directorio '/usr/src/linux-headers-5.3.0-53-generic'
feruiz@feruiz-Aspire-E5-473:~/Escritorio/EjemploModulo$
feruiz@feruiz-Aspire-E5-473:~/Escritorio/EjemploModulo$
feruiz@feruiz-Aspire-E5-473:~/Escritorio/EjemploModulo$
feruiz@feruiz-Aspire-E5-473:~/Escritorio/EjemploModulo$
feruiz@feruiz-Aspire-E5-473:~/Escritorio/EjemploModulo$
feruiz@feruiz-Aspire-E5-473:~/Escritorio/EjemploModulo$
```

Este comando sirve para insertar el módulo

> sudo insmod memo_201602434_201503984.ko

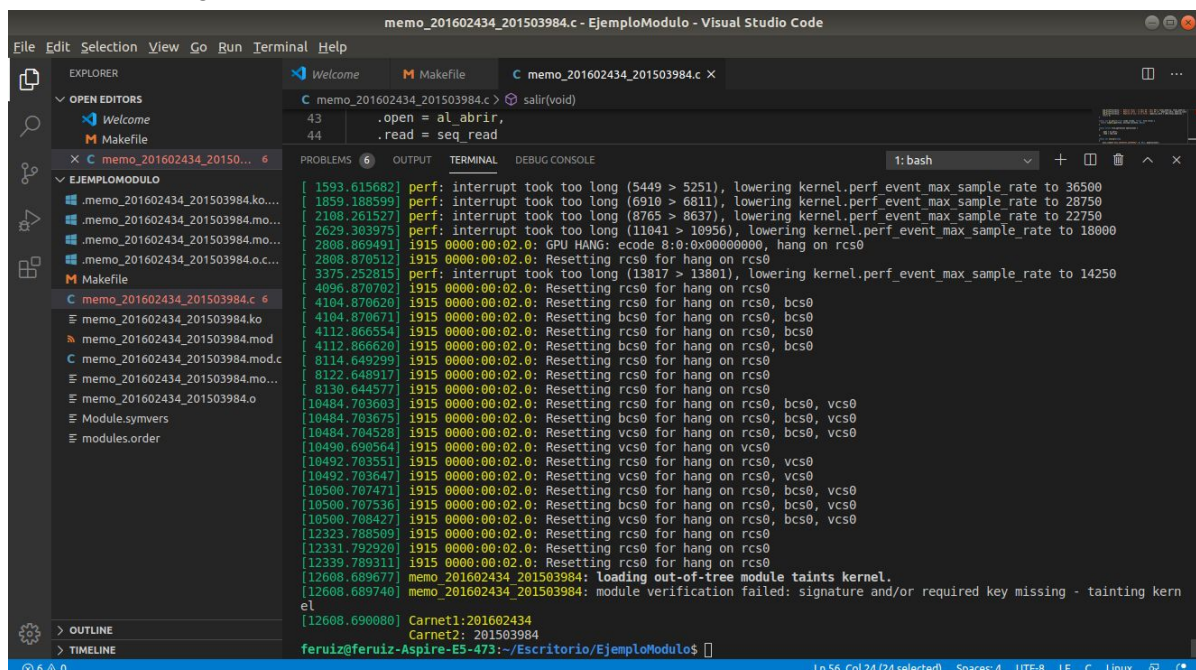
Este comando sirve para remover un módulo

> sudo rmmod memo_201602434_201503984.ko

Muestra todos los mensajes que están mostrando los módulos.

En este caso, de último se mostrarán los números de carnet (es el mensaje que se muestra al iniciar nuestro módulo).

> dmesg



The screenshot shows the Visual Studio Code editor with the file 'memo_201602434_201503984.c' open. The terminal at the bottom shows the output of 'dmesg', displaying various kernel messages including warnings about interrupt times and module loading information.

```
[ 1593.615682] perf: interrupt took too long (5449 > 5251), lowering kernel.perf_event_max_sample_rate to 36500
[ 1859.188599] perf: interrupt took too long (6910 > 6811), lowering kernel.perf_event_max_sample_rate to 28750
[ 2188.261527] perf: interrupt took too long (8765 > 8637), lowering kernel.perf_event_max_sample_rate to 22750
[ 2629.303975] perf: interrupt took too long (11941 > 10956), lowering kernel.perf_event_max_sample_rate to 18000
[ 2808.869491] 1915 0000:00:02.0: GPU HANG: ecode 8:0:0x00000000, hang on rcs0
[ 2808.870512] 1915 0000:00:02.0: Resetting rcs0 for hang on rcs0
[ 3375.252815] perf: interrupt took too long (13817 > 13801), lowering kernel.perf_event_max_sample_rate to 14250
[ 4096.870702] 1915 0000:00:02.0: Resetting rcs0 for hang on rcs0
[ 4104.870620] 1915 0000:00:02.0: Resetting rcs0 for hang on rcs0, bcs0
[ 4104.870671] 1915 0000:00:02.0: Resetting bcs0 for hang on rcs0, bcs0
[ 4112.866554] 1915 0000:00:02.0: Resetting rcs0 for hang on rcs0, bcs0
[ 4112.866620] 1915 0000:00:02.0: Resetting bcs0 for hang on rcs0, bcs0
[ 8114.649299] 1915 0000:00:02.0: Resetting rcs0 for hang on rcs0
[ 8122.648917] 1915 0000:00:02.0: Resetting rcs0 for hang on rcs0
[ 8130.644577] 1915 0000:00:02.0: Resetting rcs0 for hang on rcs0
[ 10484.703603] 1915 0000:00:02.0: Resetting rcs0 for hang on rcs0, bcs0, vcs0
[ 10484.703675] 1915 0000:00:02.0: Resetting bcs0 for hang on rcs0, bcs0, vcs0
[ 10484.704528] 1915 0000:00:02.0: Resetting vcs0 for hang on rcs0, bcs0, vcs0
[ 10490.690564] 1915 0000:00:02.0: Resetting vcs0 for hang on rcs0, vcs0
[ 10492.703551] 1915 0000:00:02.0: Resetting rcs0 for hang on rcs0, vcs0
[ 10492.703647] 1915 0000:00:02.0: Resetting vcs0 for hang on rcs0, vcs0
[ 10500.707471] 1915 0000:00:02.0: Resetting rcs0 for hang on rcs0, bcs0, vcs0
[ 10500.707536] 1915 0000:00:02.0: Resetting bcs0 for hang on rcs0, bcs0, vcs0
[ 10500.708427] 1915 0000:00:02.0: Resetting vcs0 for hang on rcs0, bcs0, vcs0
[ 12233.788509] 1915 0000:00:02.0: Resetting rcs0 for hang on rcs0
[ 12331.792920] 1915 0000:00:02.0: Resetting rcs0 for hang on rcs0
[ 12339.789311] 1915 0000:00:02.0: Resetting rcs0 for hang on rcs0
[ 12608.689677] memo_201602434_201503984: loading out-of-tree module taints kernel.
[ 12608.689740] memo_201602434_201503984: module verification failed: signature and/or required key missing - tainting kern
el
[12608.690080] Carnet1:201602434
Carnet2: 201503984
feruiz@feruiz-Aspire-E5-473:~/Escritorio/EjemploModulo$
```

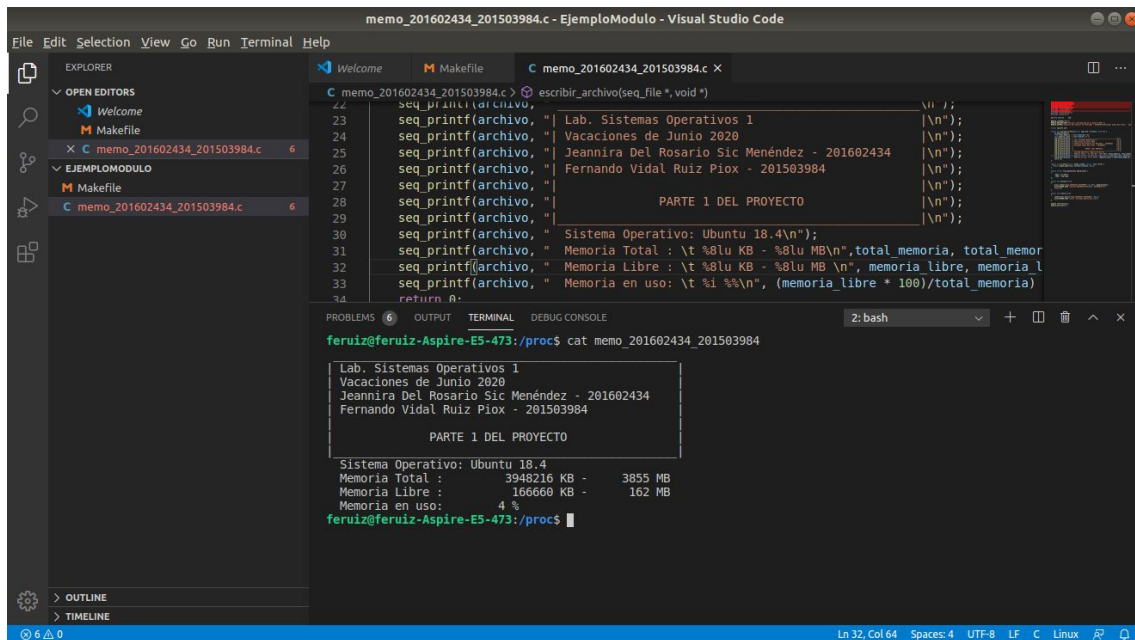

Luego abrimos una nueva terminal e ingresamos a la carpeta proc

```
> cd /proc
```

Ingresamos el siguiente comando para ver el contenido del archivo

“memo_201602434_201503984”, cabe resaltar que cada vez que ejecutamos este comando, la información al leer el archivo se actualiza.

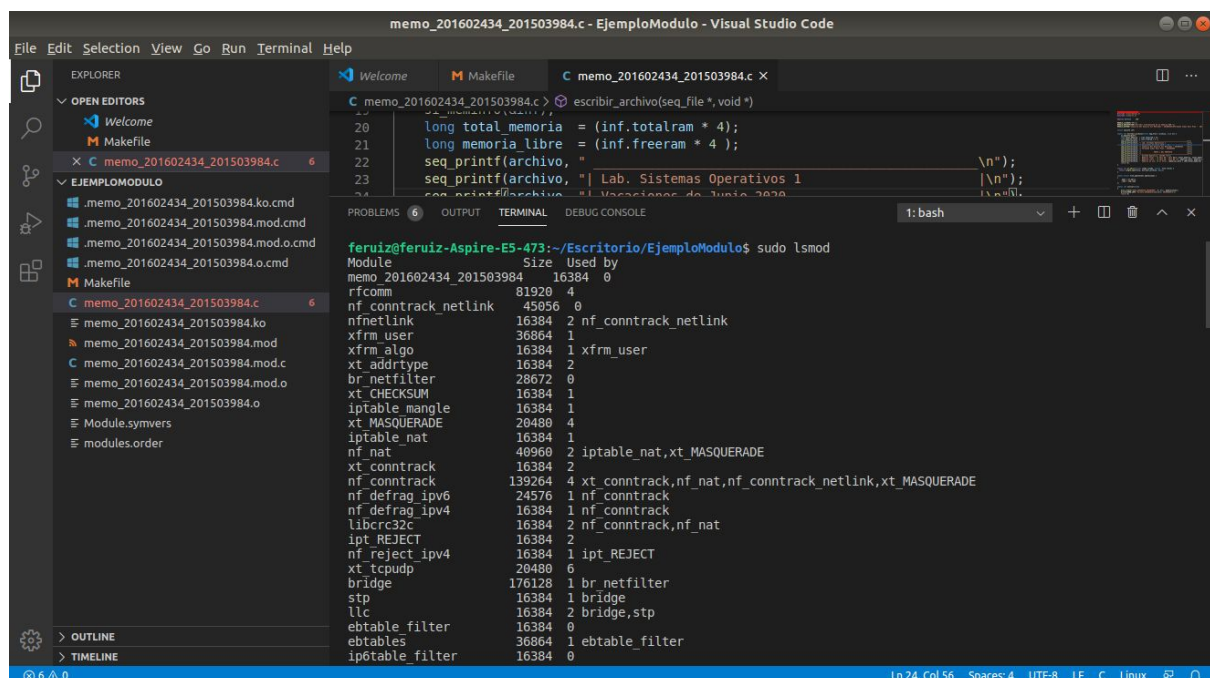
```
> cat memo_201602434_201503984
```



```
memo_201602434_201503984.c - EjemploModulo - Visual Studio Code
File Edit Selection View Go Run Terminal Help
EXPLORER
  OPEN EDITORS
    Welcome
    Makefile
    C memo_201602434_201503984.c
  EJEMPLOMODULO
    Makefile
    C memo_201602434_201503984.c
C memo_201602434_201503984.c
44 seq_printf(archivo, "Lab. Sistemas Operativos 1\n");
23 seq_printf(archivo, "Vacaciones de Junio 2020\n");
24 seq_printf(archivo, "Jeannira Del Rosario Sic Menéndez - 201602434\n");
25 seq_printf(archivo, "Fernando Vidal Ruiz Piox - 201503984\n");
26 seq_printf(archivo, "
27 seq_printf(archivo, "
28 seq_printf(archivo, "
29 seq_printf(archivo, "
30 seq_printf(archivo, "
31 seq_printf(archivo, "
32 seq_printf(archivo, "
33 seq_printf(archivo, "
34 return 0;
escribir_archivo(seq_file*, void *)
Lab. Sistemas Operativos 1
Vacaciones de Junio 2020
Jeannira Del Rosario Sic Menéndez - 201602434
Fernando Vidal Ruiz Piox - 201503984
PARTE 1 DEL PROYECTO
Sistema Operativo: Ubuntu 18.4
Memoria Total : 3948216 KB - 3855 MB
Memoria Libre : 166660 KB - 162 MB
Memoria en uso: 4 %
feruiz@feruiz-Aspire-E5-473:/proc$ cat memo_201602434_201503984
Lab. Sistemas Operativos 1
Vacaciones de Junio 2020
Jeannira Del Rosario Sic Menéndez - 201602434
Fernando Vidal Ruiz Piox - 201503984
PARTE 1 DEL PROYECTO
Sistema Operativo: Ubuntu 18.4
Memoria Total : 3948216 KB - 3855 MB
Memoria Libre : 166660 KB - 162 MB
Memoria en uso: 4 %
feruiz@feruiz-Aspire-E5-473:/proc$
```

Este comando sirve para listar todos los modulo

```
> sudo lsmod
```



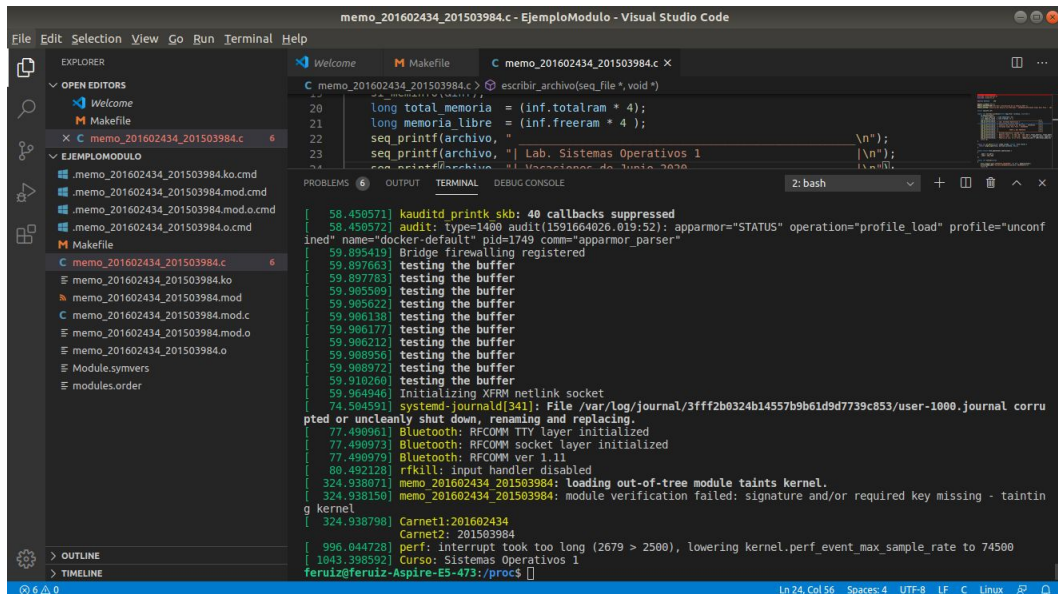
```
memo_201602434_201503984.c - EjemploModulo - Visual Studio Code
File Edit Selection View Go Run Terminal Help
EXPLORER
  OPEN EDITORS
    Welcome
    Makefile
    C memo_201602434_201503984.c
  EJEMPLOMODULO
    .memo_201602434_201503984.ko.cmd
    .memo_201602434_201503984.mod.cmd
    .memo_201602434_201503984.mod.o.cmd
    .memo_201602434_201503984.o.cmd
    Makefile
    C memo_201602434_201503984.c
    memo_201602434_201503984.ko
    memo_201602434_201503984.mod
    C memo_201602434_201503984.mod.c
    memo_201602434_201503984.mod.o
    memo_201602434_201503984.o
    Module.symvers
    modules.order
  C memo_201602434_201503984.c
20 long total_memoria = (inf.totalram * 4);
21 long memoria_libre = (inf.freeram * 4);
22 seq_printf(archivo, "\n");
23 seq_printf(archivo, "Lab. Sistemas Operativos 1\n");
24 seq_printf(archivo, "Vacaciones de Junio 2020\n");
PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE
feruiz@feruiz-Aspire-E5-473:~/EjemploModulo$ sudo lsmod
Module Size Used by
memo_201602434_201503984 16384 0
rfcomm 81920 4
nf_conntrack_netlink 45056 0
nfnetlink 16384 2 nf_conntrack_netlink
xfrm_user 36864 1
xfrm_algo 16384 1 xfrm_user
xt_addrtype 16384 2
br_netfilter 28672 0
xt_CHECKSUM 16384 1
iptable_mangle 16384 1
xt_MASQUERADE 20480 4
iptables_nat 16384 1
nf_nat 40960 2 iptable_nat,xt_MASQUERADE
xt_conntrack 16384 2
nf_conntrack 139264 4 xt_conntrack,nf_nat,nf_conntrack_netlink,xt_MASQUERADE
nf_defrag_ipv6 24576 1 nf_conntrack
nf_defrag_ipv4 16384 1 nf_conntrack
libcrc32c 16384 2 nf_conntrack,nf_nat
ipt_REJECT 16384 2
nf_reject_ipv4 16384 1 ipt_REJECT
xt_tcpudp 28480 6
bridge 176128 1 br_netfilter
stp 16384 1 bridge
llc 16384 2 bridge,stp
ebtable_filter 16384 0
ebtables 36864 1 ebtable_filter
ip6table_filter 16384 0
Ln 24, Col 56 Spaces: 4 UTF-8 LF C Linux
```

removemos el modulo

```
> sudo rmmod memo_201602434_201503984.ko
```

vemos el mensaje que muestra al ser removido el módulo.

```
> dmesg
```



The screenshot shows the Visual Studio Code interface with a terminal window open. The terminal displays the output of the `rmmod` command and the subsequent `dmesg` command. The `dmesg` output shows the kernel log, including the removal of the module and the resulting error message: "module verification failed: signature and/or required key missing - taint kernel".

```
memo_201602434_201503984.c - EjemploModulo - Visual Studio Code
File Edit Selection View Go Run Terminal Help
EXPLORER
  OPEN EDITORS
    Welcome
    Makefile
    memo_201602434_201503984.c
  EJEMPLOMODULO
    .memo_201602434_201503984.ko.cmd
    .memo_201602434_201503984.mod.cmd
    .memo_201602434_201503984.mod.o.cmd
    Makefile
    memo_201602434_201503984.c
    memo_201602434_201503984.ko
    memo_201602434_201503984.mod
    memo_201602434_201503984.mod.c
    memo_201602434_201503984.mod.o
    memo_201602434_201503984.o
    Module.symvers
    modules.order
  OUTLINE
  TIMELINE
  PROBLEMS (6) OUTPUT TERMINAL DEBUG CONSOLE
  2: bash
[ 58.450571] kauditd printk skb: 40 callbacks suppressed
[ 58.450572] audit: types=1400 audit(1591664026.019:52): apparmor="STATUS" operation="profile_load" profile="unconf
ined" name="docker-default" pid=1749 comm="apparmor_parser"
[ 59.895419] Bridge firewalling registered
[ 59.897663] testing the buffer
[ 59.897783] testing the buffer
[ 59.905589] testing the buffer
[ 59.905622] testing the buffer
[ 59.906136] testing the buffer
[ 59.906177] testing the buffer
[ 59.906212] testing the buffer
[ 59.908956] testing the buffer
[ 59.908972] testing the buffer
[ 59.910269] testing the buffer
[ 59.964946] Initializing XFRM netlink socket
[ 74.504591] systemd-journald[341]: File /var/log/journal/3ff2b0324b14557b9b61d9d7739c853/user-1000.journal corru
pted or uncleanly shut down, renaming and replacing.
[ 77.490901] Bluetooth: RFCOMM TTY layer initialized
[ 77.490973] Bluetooth: RFCOMM socket layer initialized
[ 77.490979] Bluetooth: RFCOMM ver 1.11
[ 80.492128] rkill: input handler disabled
[ 324.938071] memo_201602434_201503984: loading out-of-tree module taints kernel.
[ 324.938190] memo_201602434_201503984: module verification failed: signature and/or required key missing - taint
ing kernel
[ 324.938798] Carnet1:201602434
[ 324.938798] Carnet2:201503984
[ 996.044728] perf: interrupt took too long (2679 > 2500), lowering kernel.perf_event_max_sample_rate to 74500
[1043.398592] Curso: Sistemas Operativos 1
feruiz@feruiz-Aspire-E5-473:/proc$
```

Módulo de CPU (task_struct)

crear dos archivos

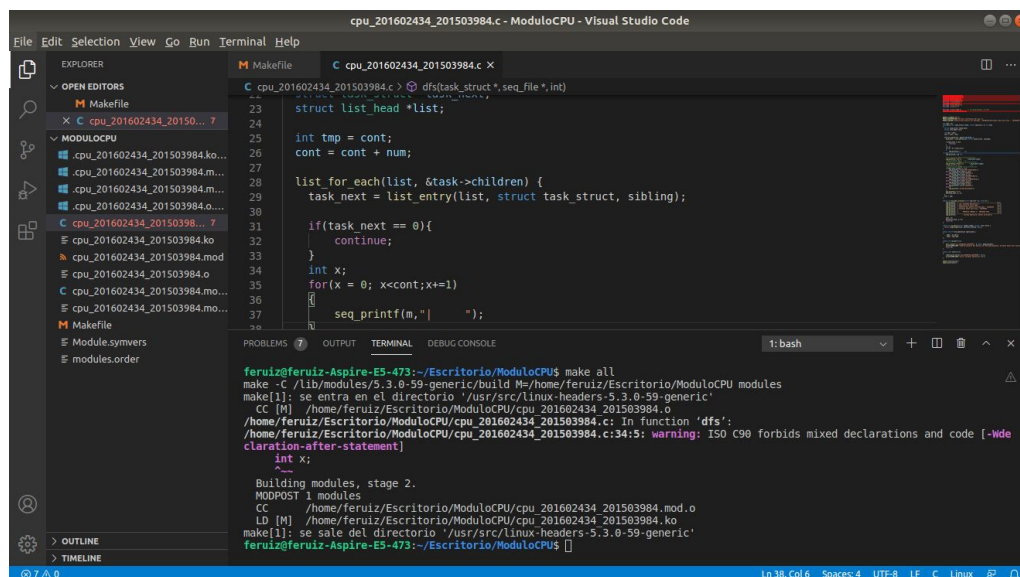
Makefile

cpu_201602434_201503984.c

Luego se utilizan los siguiente comandos:

Para crear los archivos necesarios. (agregaré 11 archivos más pero el que nos importará es el archivo .ko)

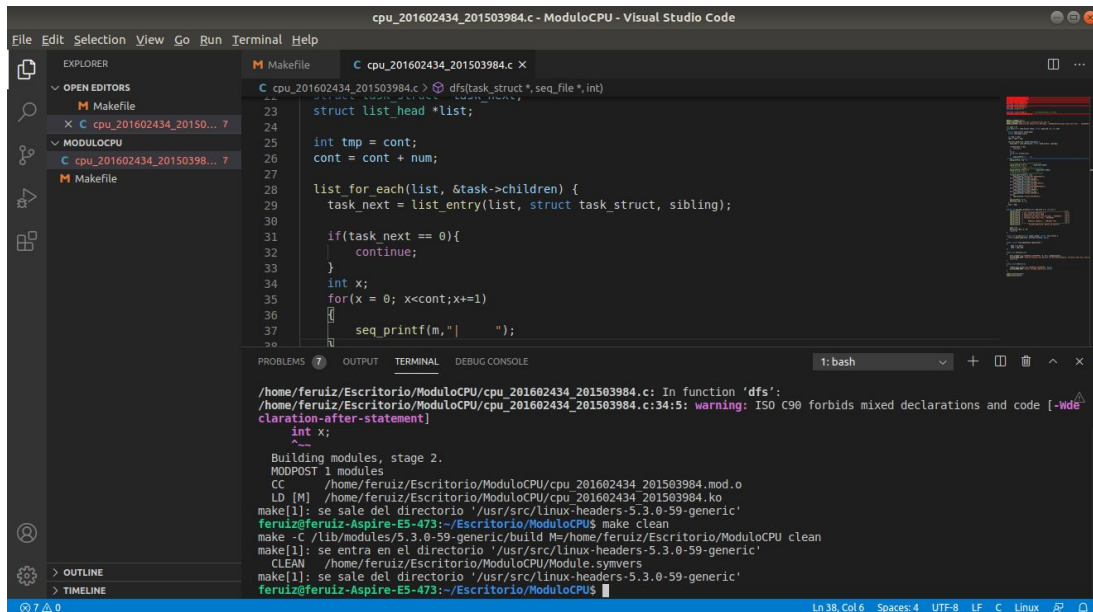
```
> make all
```



The screenshot shows the Visual Studio Code interface with a terminal window open. The terminal displays the output of the `make` command, showing the compilation of the module. The output includes the command `make -C /lib/modules/5.3.0-59-generic/build M=/home/feruiz/Escritorio/ModuloCPU modules` and the resulting compilation steps, including the creation of the `cpu_201602434_201503984.ko` file.

```
cpu_201602434_201503984.c - ModuloCPU - Visual Studio Code
File Edit Selection View Go Run Terminal Help
EXPLORER
  OPEN EDITORS
    Makefile
    cpu_201602434_201503984.c
  MODULOCPU
    .cpu_201602434_201503984.ko.cmd
    .cpu_201602434_201503984.mod.cmd
    .cpu_201602434_201503984.mod.o.cmd
    Makefile
    cpu_201602434_201503984.c
    cpu_201602434_201503984.ko
    cpu_201602434_201503984.mod
    cpu_201602434_201503984.mod.c
    cpu_201602434_201503984.mod.o
    cpu_201602434_201503984.o
    Module.symvers
    modules.order
  OUTLINE
  TIMELINE
  PROBLEMS (7) OUTPUT TERMINAL DEBUG CONSOLE
  1: bash
feruiz@feruiz-Aspire-E5-473:~/Escritorio/ModuloCPU$ make all
make -C /lib/modules/5.3.0-59-generic/build M=/home/feruiz/Escritorio/ModuloCPU modules
make[1]: se entra en el directorio '/usr/src/linux-headers-5.3.0-59-generic'
CC [M] /home/feruiz/Escritorio/ModuloCPU/cpu_201602434_201503984.o
/home/feruiz/Escritorio/ModuloCPU/cpu_201602434_201503984.c: In function 'dfs':
/home/feruiz/Escritorio/ModuloCPU/cpu_201602434_201503984.c:34:5: warning: ISO C90 forbids mixed declarations and code [-Wde
claration-after-statement]
     int x;
     ^
Building modules, stage 2.
MODPOST 1 modules
CC /home/feruiz/Escritorio/ModuloCPU/cpu_201602434_201503984.mod.o
LD [M] /home/feruiz/Escritorio/ModuloCPU/cpu_201602434_201503984.ko
make[1]: se sale del directorio '/usr/src/linux-headers-5.3.0-59-generic'
feruiz@feruiz-Aspire-E5-473:~/Escritorio/ModuloCPU$
```

Para limpiar los archivos que se crearon al ejecutar el comando anterior
> make clean



```
File Edit Selection View Go Run Terminal Help
cpu_201602434_201503984.c - ModuloCPU - Visual Studio Code

EXPLORER
  Makefile
  cpu_201602434_201503984.c
  ModuloCPU
  cpu_201602434_201503984.mod.o
  Makefile

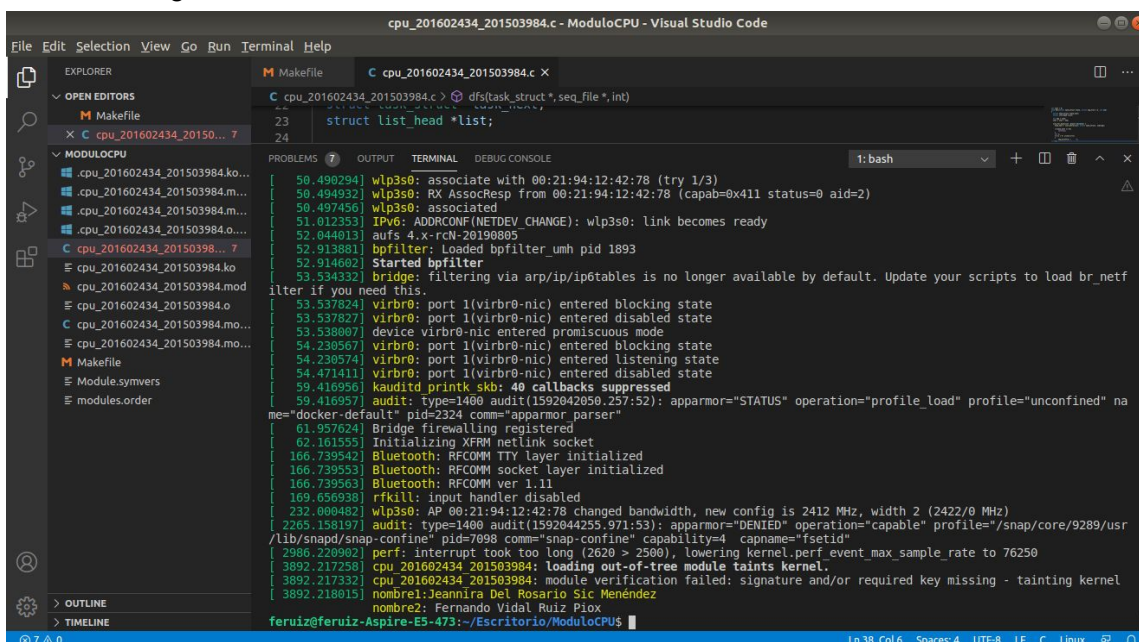
OPEN EDITORS
  Makefile
  cpu_201602434_201503984.c
  ModuloCPU
  cpu_201602434_201503984.mod.o
  Makefile

cpu_201602434_201503984.c
23 struct list_head *list;
24
25 int tmp = cont;
26 cont = cont + num;
27
28 list_for_each(list, &task->children) {
29     task_next = list_entry(list, struct task_struct, sibling);
30
31     if(task_next == 0){
32         continue;
33     }
34     int x;
35     for(x = 0; x<cont;x+=1)
36         seq_printf(m,"| ");
37 }
38
39 PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE
1: bash
/home/feruiz/Escritorio/ModuloCPU/cpu_201602434_201503984.c: In function 'dfs':
/home/feruiz/Escritorio/ModuloCPU/cpu_201602434_201503984.c:34:5: warning: ISO C90 forbids mixed declarations and code [-Wdeclaration-after-statement]
    int x;
    ^
Building modules, stage 2.
MODPOST 1 modules
CC      /home/feruiz/Escritorio/ModuloCPU/cpu_201602434_201503984.mod.o
LD [M]  /home/feruiz/Escritorio/ModuloCPU/cpu_201602434_201503984.ko
make[1]: se sale del directorio '/usr/src/linux-headers-5.3.0-59-generic'
feruiz@feruiz-Aspire-E5-473:~/Escritorio/ModuloCPU$ make clean
make -C /lib/modules/5.3.0-59-generic/build M=/home/feruiz/Escritorio/ModuloCPU clean
make[1]: se entra en el directorio '/usr/src/linux-headers-5.3.0-59-generic'
CLEAN /home/feruiz/Escritorio/ModuloCPU/Module.symvers
make[1]: se sale del directorio '/usr/src/linux-headers-5.3.0-59-generic'
feruiz@feruiz-Aspire-E5-473:~/Escritorio/ModuloCPU$
```

Este comando sirve para insertar el módulo
> sudo insmod cpu_201602434_201503984.ko

Este comando sirve para remover un módulo
> sudo rmmod cpu_201602434_201503984.ko

Después ejecutaremos el siguiente comando, para mostrar el mensaje del modulo.
En este caso,de último se mostrarán los nombres de los integrantes del grupo (es el mensaje que se muestra al iniciar nuestro módulo).
> dmesg



```
File Edit Selection View Go Run Terminal Help
cpu_201602434_201503984.c - ModuloCPU - Visual Studio Code

EXPLORER
  Makefile
  cpu_201602434_201503984.c
  ModuloCPU
  cpu_201602434_201503984.mod.o
  Makefile

OPEN EDITORS
  Makefile
  cpu_201602434_201503984.c
  ModuloCPU
  cpu_201602434_201503984.mod.o
  Makefile

cpu_201602434_201503984.c
23 struct list_head *list;
24
25 int tmp = cont;
26 cont = cont + num;
27
28 list_for_each(list, &task->children) {
29     task_next = list_entry(list, struct task_struct, sibling);
30
31     if(task_next == 0){
32         continue;
33     }
34     int x;
35     for(x = 0; x<cont;x+=1)
36         seq_printf(m,"| ");
37 }
38
39 PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE
1: bash
[ 50.490294] wlp3s0: associate with 00:21:94:12:42:78 (try 1/3)
[ 50.494932] wlp3s0: RX AssocResp from 00:21:94:12:42:78 (capab=0x411 status=0 aid=2)
[ 50.497456] wlp3s0: associated
[ 51.012353] IPVS: ADDRCONF(NETDEV_CHANGE): wlp3s0: link becomes ready
[ 52.044013] aufs: 4x-rch-20190805
[ 52.913881] bpfILTER: Loaded bpfILTER_umh pid 1893
[ 52.914602] Started bpfILTER
[ 53.534332] bridge: filtering via arp/ip6tables is no longer available by default. Update your scripts to load br_netfilter if you need this.
[ 53.537824] virbr0: port 1(virbr0-nic) entered blocking state
[ 53.537827] virbr0: port 1(virbr0-nic) entered disabled state
[ 53.538007] device virbr0-nic entered promiscuous mode
[ 54.230567] virbr0: port 1(virbr0-nic) entered blocking state
[ 54.230574] virbr0: port 1(virbr0-nic) entered listening state
[ 54.471411] virbr0: port 1(virbr0-nic) entered disabled state
[ 59.416956] kauditd: printk skb: 40 callbacks suppressed
[ 59.416957] audit: type=1400 audit(1592042050.257:52): apparmor="STATUS" operation="profile_load" profile="unconfined" name="docker-default" pid=2324 comm="apparmor_parser"
[ 61.957624] Bridge firewalling registered
[ 62.161555] Initializing XFRM netlink socket
[ 166.739542] Bluetooth: RFCOMM TTY layer initialized
[ 166.739553] Bluetooth: RFCOMM socket layer initialized
[ 166.739563] Bluetooth: RFCOMM ver 1.11
[ 169.656938] rkill: input handler disabled
[ 232.000482] wlp3s0: AP 00:21:94:12:42:78 changed bandwidth, new config is 2412 MHz, width 2 (2422/0 MHz)
[ 235.158197] audit: type=1400 audit(1592044255.971:53): apparmor="DENIED" operation="capable" profile="/snap/core/9289/usr/lib/snapd/snap-confine" pid=7698 comm="snap-confine" capability=4 capname="fsetid"
[ 2986.220902] perf: interrupt took too long (2620 > 2500), lowering kernel.perf_event_max_sample_rate to 76250
[ 3892.217258] cpu 201602434_201503984: loading out-of-tree module taints kernel.
[ 3892.217332] cpu 201602434_201503984: module verification failed: signature and/or required key missing - tainting kernel
[ 3892.218015] nombre1:Jeannira Del Rosario Sic Menéndez
[ 3892.218015] nombre2: Fernando Vidal Ruiz Piox
feruiz@feruiz-Aspire-E5-473:~/Escritorio/ModuloCPU$
```

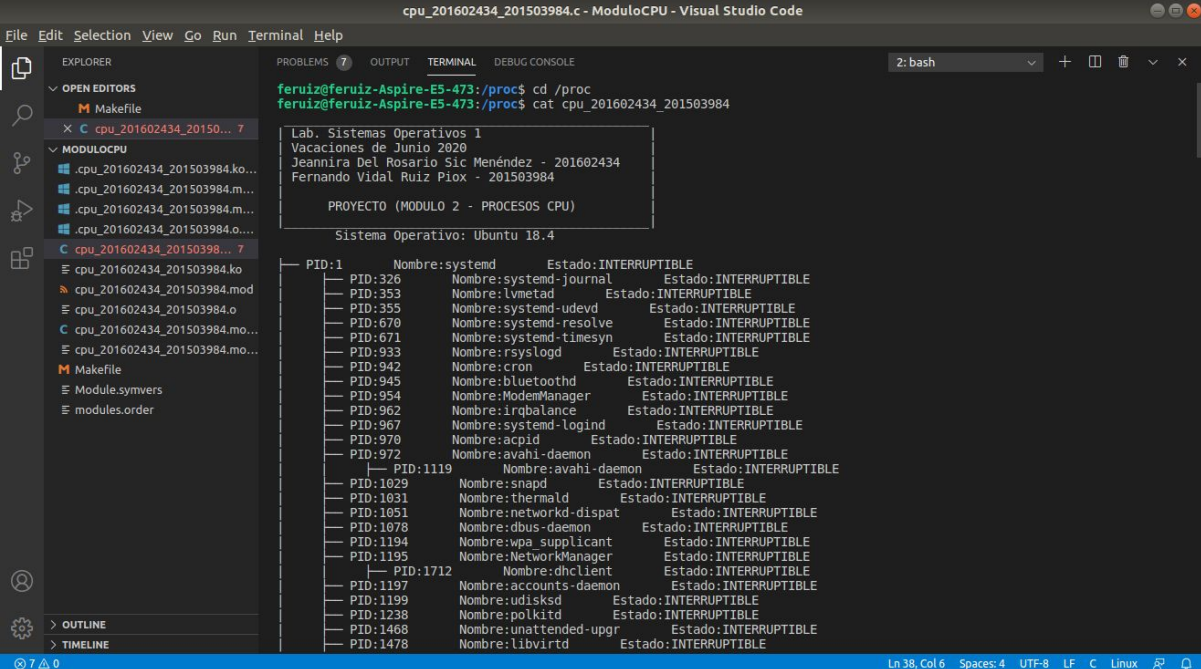

luego abrimos una nueva terminal e ingresamos a la carpeta proc

```
> cd /proc
```

Ingresamos el siguiente comando para ver el contenido del archivo

“cpu_201602434_201503984”, cabe resaltar que cada vez que ejecutamos este comando, la información al leer el archivo se actualiza.

```
> cat cpu_201602434_201503984
```



```
cpu_201602434_201503984.c - ModuloCPU - Visual Studio Code
File Edit Selection View Go Run Terminal Help
EXPLORER
  OPEN EDITORS
    Makefile
  MODULO CPU
    .cpu_201602434_201503984.ko...
    .cpu_201602434_201503984.m...
    .cpu_201602434_201503984.o...
    .cpu_201602434_201503984.o...
    C cpu_201602434_20150398... 7
    E cpu_201602434_201503984.ko...
    S cpu_201602434_201503984.mod
    E cpu_201602434_201503984.o...
    C cpu_201602434_201503984.mo...
    E cpu_201602434_201503984.mo...
    Makefile
    Module.symvers
    modules.order
  OUTLINE
  TIMELINE
  PROBLEMS 7 OUTPUT TERMINAL DEBUG CONSOLE
  2: bash
feruiz@feruiz-Aspire-E5-473:/proc$ cd /proc
feruiz@feruiz-Aspire-E5-473:/proc$ cat cpu_201602434_201503984
Lab. Sistemas Operativos 1
Vacaciones de Junio 2020
Jeannira Del Rosario Sic Menéndez - 201602434
Fernando Vidal Ruiz Piox - 201503984

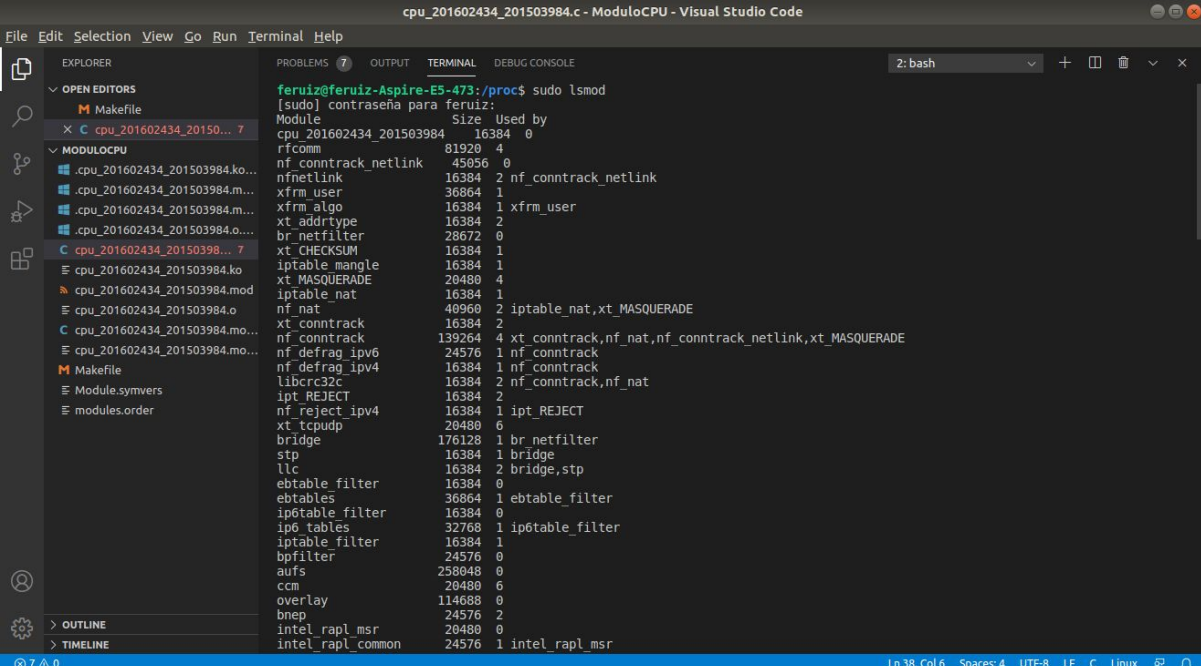
PROYECTO (MODULO 2 - PROCESOS CPU)

Sistema Operativo: Ubuntu 18.4

PID:1      Nombre:systemd      Estado:INTERRUPTIBLE
PID:326    Nombre:systemd-journal Estado:INTERRUPTIBLE
PID:353    Nombre:lvmetad      Estado:INTERRUPTIBLE
PID:355    Nombre:systemd-udev  Estado:INTERRUPTIBLE
PID:670    Nombre:systemd-resolve Estado:INTERRUPTIBLE
PID:671    Nombre:systemd-timesyn Estado:INTERRUPTIBLE
PID:933    Nombre:rsyslogd      Estado:INTERRUPTIBLE
PID:942    Nombre:cron          Estado:INTERRUPTIBLE
PID:945    Nombre:bluetoothd    Estado:INTERRUPTIBLE
PID:954    Nombre:ModemManager  Estado:INTERRUPTIBLE
PID:962    Nombre:irqbalance    Estado:INTERRUPTIBLE
PID:967    Nombre:systemd-logind Estado:INTERRUPTIBLE
PID:970    Nombre:acpid          Estado:INTERRUPTIBLE
PID:972    Nombre:avahi-daemon  Estado:INTERRUPTIBLE
PID:1119   Nombre:avahi-daemon  Estado:INTERRUPTIBLE
PID:1029   Nombre:snapd          Estado:INTERRUPTIBLE
PID:1031   Nombre:thermald      Estado:INTERRUPTIBLE
PID:1051   Nombre:networkd-dispat Estado:INTERRUPTIBLE
PID:1078   Nombre:dbus-daemon   Estado:INTERRUPTIBLE
PID:1194   Nombre:wpa_supplicant Estado:INTERRUPTIBLE
PID:1195   Nombre:NetworkManager Estado:INTERRUPTIBLE
PID:1197   Nombre:accounts-daemon Estado:INTERRUPTIBLE
PID:1199   Nombre:udisksd       Estado:INTERRUPTIBLE
PID:1238   Nombre:polkitd       Estado:INTERRUPTIBLE
PID:1468   Nombre:unattended-upgr Estado:INTERRUPTIBLE
PID:1478   Nombre:libvirtd      Estado:INTERRUPTIBLE
```

Este comando sirve para listar todos los modulo.

```
> sudo lsmod
```



```
cpu_201602434_201503984.c - ModuloCPU - Visual Studio Code
File Edit Selection View Go Run Terminal Help
EXPLORER
  OPEN EDITORS
    Makefile
  MODULO CPU
    .cpu_201602434_201503984.ko...
    .cpu_201602434_201503984.m...
    .cpu_201602434_201503984.m...
    .cpu_201602434_201503984.o...
    C cpu_201602434_20150398... 7
    E cpu_201602434_201503984.ko...
    S cpu_201602434_201503984.mod
    E cpu_201602434_201503984.o...
    C cpu_201602434_201503984.mo...
    E cpu_201602434_201503984.mo...
    Makefile
    Module.symvers
    modules.order
  OUTLINE
  TIMELINE
  PROBLEMS 7 OUTPUT TERMINAL DEBUG CONSOLE
  2: bash
feruiz@feruiz-Aspire-E5-473:/proc$ sudo lsmod
[sudo] contraseña para feruiz:
Module                               Size Used by
nf_conntrack_netlink                45056  0
nf_conntrack                        16384  0
nfnetlink                           16384  2 nf_conntrack_netlink
xfrm_user                           36864  1
xfrm_algo                           16384  1 xfrm_user
xt_addrtype                         16384  2
br_netfilter                        28672  0
xt_CHECKSUM                         16384  1
iptable_mangle                      16384  1
xt_MASQUERADE                       20480  4
iptable_nat                         16384  1
nf_nat                              40960  2 iptable_nat,xt_MASQUERADE
xt_conntrack                        139264  4 xt_conntrack,nf_nat,nf_conntrack_netlink,xt_MASQUERADE
nf_defrag_ipv6                      24576  1 nf_conntrack
nf_defrag_ipv4                      16384  1 nf_conntrack
libcrc32c                           16384  2 nf_conntrack,nf_nat
ipt_REJECT                          16384  2
nf_reject_ipv4                      16384  1 ipt_REJECT
xt_tcpudp                           20480  6
bridge                              176128  1 br_netfilter
stp                                 16384  1 bridge
llc                                 16384  2 bridge,stp
ebtable_filter                      16384  0
ebtables                            36864  1 ebtable_filter
ip6table_filter                     16384  0
ip6_tables                          32768  1 ip6table_filter
iptable_filter                      16384  1
bpfILTER                            24576  0
aufs                                258048  0
ccm                                 20480  6
overlay                             114688  0
bnep                                24576  2
intel_rapl_msr                      20480  0
intel_rapl_common                   24576  1 intel_rapl_msr
```

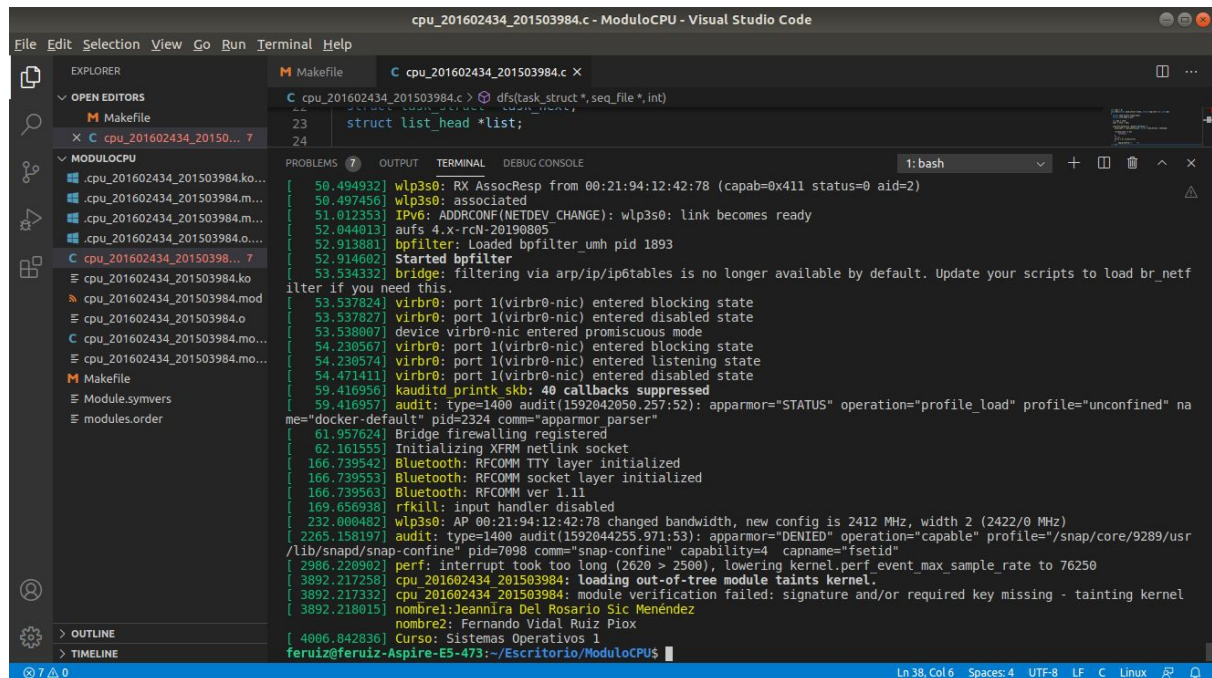

removemos el modulo

```
> sudo rmmod memo_201602434_201503984.ko
```

Después ejecutaremos el siguiente comando, para mostrar el mensaje del modulo.

En este caso,de último se mostrarán el nombre del curso (es el mensaje que se muestra al deshabilitar nuestro módulo).

```
> dmesg
```



The screenshot shows the Visual Studio Code interface with a terminal window open. The terminal displays the output of the `dmesg` command, showing various kernel messages. The messages include:

- `wlp3s0: RX AssocResp from 00:21:94:12:42:78 (capab=0x411 status=0 aid=2)`
- `wlp3s0: associated`
- `IPv6: ADDRCONF(NETDEV_CHANGE): wlp3s0: link becomes ready`
- `aufs 4.x-rcN-20190805`
- `bpfilter: Loaded bpfilter_umh pid 1893`
- `Started bpfilter`
- `bridge: filtering via arp/ip6tables is no longer available by default. Update your scripts to load br_netf`
- `filter if you need this.`
- `virbr0: port 1(virbr0-nic) entered blocking state`
- `virbr0: port 1(virbr0-nic) entered disabled state`
- `device virbr0-nic entered promiscuous mode`
- `virbr0: port 1(virbr0-nic) entered blocking state`
- `virbr0: port 1(virbr0-nic) entered listening state`
- `virbr0: port 1(virbr0-nic) entered disabled state`
- `kauditd_printk_skb: 40 callbacks suppressed`
- `audit: type=1400 audit(1592042050.257:52): apparmor="STATUS" operation="profile_load" profile="unconfined" na`
- `me="docker-default" pid=2324 comm="apparmor parser"`
- `61.957624] Bridge firewalling registered`
- `62.161555] Initializing XFRM netlink socket`
- `166.739542] Bluetooth: RFCOMM TTY layer initialized`
- `166.739553] Bluetooth: RFCOMM socket layer initialized`
- `166.739563] Bluetooth: RFCOMM ver 1.11`
- `169.656938] rkill: input handler disabled`
- `232.000482] wlp3s0: AP 00:21:94:12:42:78 changed bandwidth, new config is 2412 MHz, width 2 (2422/0 MHz)`
- `2265.158197] audit: type=1400 audit(1592044255.971:53): apparmor="DENIED" operation="capable" profile="/snap/core/9289/usr`
- `/lib/snapd/snap-confine" pid=7098 comm="snap-confine" capability=4 capname="fsetid"`
- `2986.220902] perf: interrupt took too long (2620 > 2500), lowering kernel.perf_event_max_sample_rate to 76250`
- `3892.217258] cpu_201602434_201503984: loading out-of-tree module taints kernel.`
- `cpu_201602434_201503984: module verification failed: signature and/or required key missing - tainting kernel`
- `nombre1: Jeannira Del Rosario Sic Menéndez`
- `nombre2: Fernando Vidal Ruiz Piox`
- `4006.842836] Curso: Sistemas Operativos I`

The terminal window is titled `cpu_201602434_201503984.c - ModuloCPU - Visual Studio Code`. The status bar at the bottom shows `Ln 38, Col 6`, `Spaces: 4`, `UTF-8`, `LF`, `C`, `Linux`, and a search icon.

Segunda Parte del Proyecto

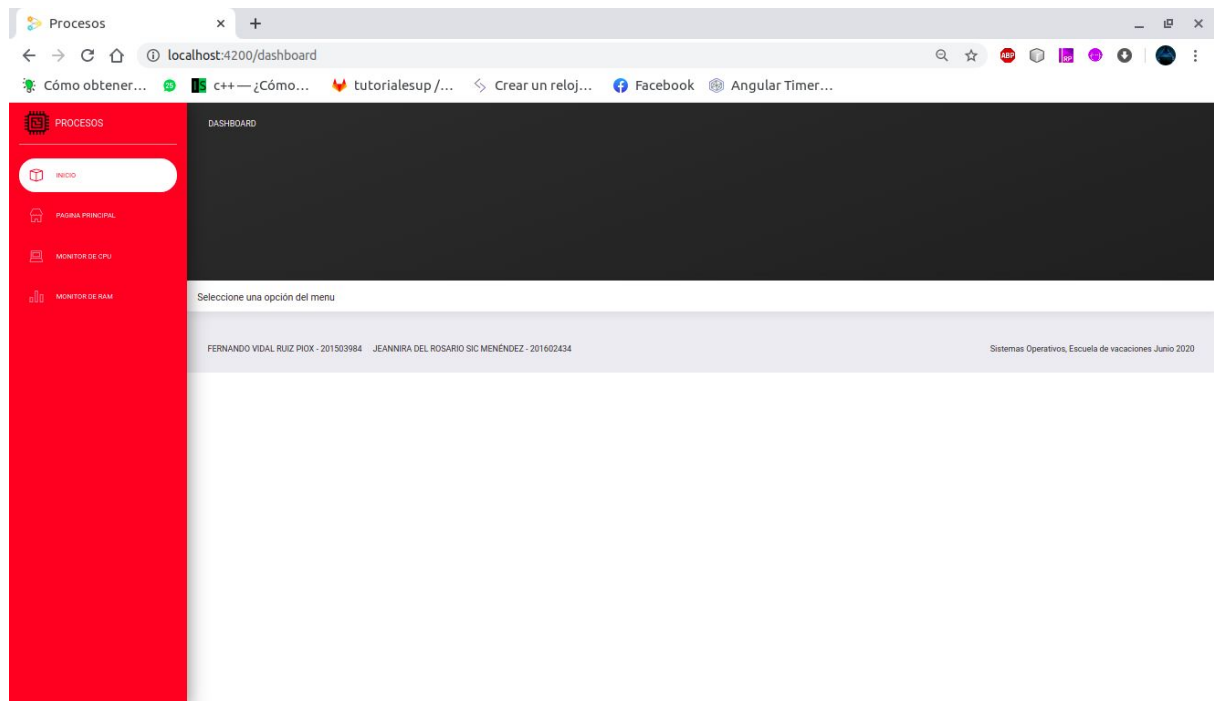
Aplicación WEB

La aplicación web permite visualizar gráficas dinámicas que muestren el uso del CPU y de la memoria RAM del servidor.

La aplicación web permite mostrar la información básica de los procesos que se ejecutan y permite terminar los procesos(kill) que se encuentran en ejecución.

Página de Inicio

Muestra la información general de los creadores del proyecto.



Página principal

Muestra tabulada todos los procesos que están siendo ejecutados en el servidor, así como un resumen general de los procesos. La información general que se muestra es la siguiente:

Procesos en ejecución	Número de procesos en estado de ejecución (running)
Procesos suspendidos	Número de procesos en estado de suspendido (sleeping)
Procesos detenidos	Número de procesos en estado de detenido (stoped)
Procesos Zombie	Número de procesos zombie
Total de procesos	Número total de procesos

Además muestra de manera tabulada la siguiente información de cada proceso:

PID	Identificador del proceso
Nombre	Nombre del proceso
Usuario	Usuario que ejecutó el proceso

Estado
%RAM

Estado en el que se encuentra el proceso
Porcentaje de utilización de RAM por el proceso

En esta sección, se encuentra un botón el cual desarrolla la función de terminar (kill) al proceso seleccionado en la tabla ya descrita.

The screenshot shows the 'Procesos' application interface. On the left is a red sidebar with navigation options: PROCESOS, INICIO, PAGINA PRINCIPAL (selected), MONITOR DE CPU, and MONITOR DE RAM. The main content area is titled 'Página principal' and contains a 'Resumen general' (General Summary) section with the following data:

Procesos en ejecución	Procesos suspendidos	Procesos detenidos	Procesos Zombie	Total de procesos
1	226	0	0	284

Below the summary is a 'Listado de procesos' (Process List) section with a 'Filtrar' (Filter) input. The list is a table with the following columns: PID, Nombre, Usuario, Estado, %RAM, and Acción. The first five rows of the table are:

PID	Nombre	Usuario	Estado	%RAM	Acción
1	systemd	root	S	5.716816%	Kill
10	ksoftirqd/0	root	S	0.000000%	Kill
1029	snaped	root	S	33.164278%	Kill
1031	thermald	root	S	4.736220%	Kill
1051	networkd-dispat	root	S	4.499151%	Kill

At the bottom of the table, there is a 'Items per page' dropdown set to 5 and a pagination indicator '1 - 5 of 285'.

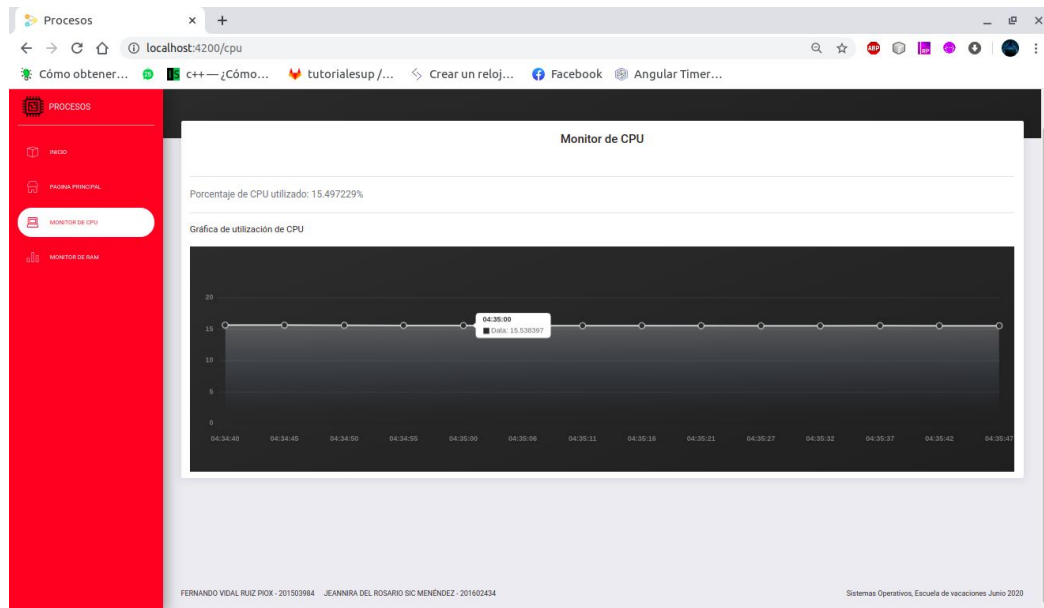
Se muestra un árbol de procesos dinámico, es decir, se muestra una lista de procesos que permita seleccionar uno y desplegar a todos sus hijos con PID y nombre.

The screenshot shows the 'Procesos' application interface with the 'Árbol de procesos' (Process Tree) section selected. The top section shows the details for PID 972, 'avahi-daemon', owned by 'avahi', in state 'S', using 1.196896% RAM, with a 'Kill' button. Below this is the 'Árbol de procesos' section, which displays a hierarchical tree of processes. The root of the tree is 'PID: 1, Nombre: systemd'. It has several children, including 'PID: 2028, Nombre: dnsmasq', 'PID: 2090, Nombre: dockerd', 'PID: 2092, Nombre: whoopsie', 'PID: 2110, Nombre: kerneloops', 'PID: 2120, Nombre: kerneloops', 'PID: 2386, Nombre: upowerd', 'PID: 2699, Nombre: rtkit-daemon', 'PID: 2722, Nombre: ibus-x11', 'PID: 2992, Nombre: bolt', 'PID: 3002, Nombre: packagekitd', 'PID: 3192, Nombre: colord', 'PID: 326, Nombre: systemd-journal', and 'PID: 327, Nombre: systemd-logind'. The 'PID: 2715, Nombre: xdg-permission' process has children 'PID: 2725, Nombre: ibus-portals', 'PID: 2890, Nombre: dnsmasq', 'PID: 3582, Nombre: [sd-pam]', 'PID: 3610, Nombre: dbus-daemon', 'PID: 3712, Nombre: at-spi-bus-launcher', 'PID: 3719, Nombre: at-spi2-regist', 'PID: 3744, Nombre: glibd', 'PID: 3748, Nombre: glibd-fuse', 'PID: 3790, Nombre: ibus-panel', 'PID: 3791, Nombre: xdg-permission', 'PID: 3801, Nombre: gnome-shell', 'PID: 3810, Nombre: glibd-udisks2', and 'PID: 3811, Nombre: evolution'. The 'PID: 3712, Nombre: at-spi-bus-launcher' process has children 'PID: 3717, Nombre: dbus-daemon', 'PID: 4028, Nombre: glibd-trash', 'PID: 6471, Nombre: glibd-network', and 'PID: 6568, Nombre: glibd-dnsd'. The footer of the application shows the text 'FERNANDO VIDAL RUIZ PLOX - 201503984 JEANNIRA DEL ROSARIO SIC MENÉNDEZ - 201602434' and 'Sistemas Operativos, Escuela de vacaciones Junio 2020'.

Monitor de CPU

El monitor de CPU debe muestra la información del consumo de CPU del servidor, en el cual se visualiza la siguiente información:

- Porcentaje de CPU utilizado
- Gráfica de utilización de CPU en tiempo real



Monitor de RAM

El monitor de memoria RAM es similar al de CPU, muestra la información del consumo de RAM del servidor, en el cual se visualiza la siguiente información:

- Total de memoria RAM del servidor (en MB)
- Total de memoria RAM consumida (en MB)
- Porcentaje de consumo de RAM

