

Systems Advanced Linux

Process Management



**DE HOGESCHOOL
MET HET NETWERK**

Elfde-Liniestraat 24, 3500 Hasselt, www.pxl.be



Terminology

- **process**

Programma of commando dat een bepaalde job uitvoert

- **PID**

Ieder proces heeft een process id

Dit is een uniek nummer (tussen 0 en 65535)

- **PPID**

Ieder proces heeft een parent process, met een parent PID

Een child process wordt gestart door zijn parent process

Terminology

- **systemd** Het systemd proces, heeft als PID 1, wordt gestart door de kernel zelf en heeft geen parent process
→ wordt soms nog aangeduid met **init**
- **kill** Als een proces stopt, sterft het proces.
Als je een proces wil stoppen, moet je het “killen”
- **daemon** Een proces dat start bij het opstarten van je systeem en vervolgens continu blijft draaien

Basic Process Management

- **\$\$** Shell parameter: bevat het **huidig process ID**
- **\$PPID** Shell variabele: bevat het **parent ID**

```
student@ubuntu-server-2:~$ echo $$ $PPID
1354 1353
student@ubuntu-server-2:~$
```

1354 is het process ID van bash

1353 is het process ID van de sshd connectie naar de VM

Basic Process Management

- **pidof** Vind alle process id's op naam.

```
student@ubuntu-server-2:~$ pidof sshd
1353 1303 939
student@ubuntu-server-2:~$ bash
student@ubuntu-server-2:~$ bash
student@ubuntu-server-2:~$ bash
student@ubuntu-server-2:~$ pidof bash
1711 1705 1699 1354
student@ubuntu-server-2:~$ exit
exit
student@ubuntu-server-2:~$ exit
exit
student@ubuntu-server-2:~$ exit
exit
student@ubuntu-server-2:~$ pidof sshd
1353 1303 939
student@ubuntu-server-2:~$ pidof bash
1354
student@ubuntu-server-2:~$
```



Basic Process Management

- **parent and child**

leder process (buiten systemd) heeft een parent process.

```
student@ubuntu-server-2:~$ echo $$ $PPID
1354 1353
student@ubuntu-server-2:~$ bash
student@ubuntu-server-2:~$ echo $$ $PPID
1720 1354
student@ubuntu-server-2:~$ exit
exit
student@ubuntu-server-2:~$ echo $$ $PPID
1354 1353
student@ubuntu-server-2:~$
```

Basic Process Management

PS(1)

User Commands

PS(1)

NAME

ps - report a snapshot of the current processes.

man ps

```
student@ubuntu-server-2:~$ man ps
```

```
student@ubuntu-server-2:~$ ps ax | head -4
```

PID	TTY	STAT	TIME	COMMAND
1	?	Ss	0:03	/sbin/init
2	?	S	0:00	[kthreadd]
3	?	I<	0:00	[rcu_gp]

```
student@ubuntu-server-2:~$ ps --pid 1 -o pid,cmd,comm
```

PID	CMD	COMMAND
1	/sbin/init	systemd

```
student@ubuntu-server-2:~$
```

Basic Process Management

- Werken met **ps**

```
student@ubuntu-server-2:~$ echo $$ $PPID
1354 1353
student@ubuntu-server-2:~$ bash
student@ubuntu-server-2:~$ echo $$ $PPID
1767 1354
student@ubuntu-server-2:~$ bash
student@ubuntu-server-2:~$ echo $$ $PPID
1773 1767
student@ubuntu-server-2:~$ ps f
  PID TTY          STAT       TIME COMMAND
 1354 pts/0        Ss          0:00 -bash
 1767 pts/0        S           0:00 \_ bash
 1773 pts/0        S           0:00 \_ bash
 1779 pts/0        R+          0:00 \_ ps f
student@ubuntu-server-2:~$ exit
exit
student@ubuntu-server-2:~$ ps f
  PID TTY          STAT       TIME COMMAND
 1354 pts/0        Ss          0:00 -bash
 1767 pts/0        S           0:00 \_ bash
 1781 pts/0        R+          0:00 \_ ps f
student@ubuntu-server-2:~$ exit
exit
student@ubuntu-server-2:~$ ps f
  PID TTY          STAT       TIME COMMAND
 1354 pts/0        Ss          0:00 -bash
 1782 pts/0        R+          0:00 \_ ps f
student@ubuntu-server-2:~$
```

met **ps f -f** zie je er ook de PID en PPID bij

Basic Process Management

pgrep

processen op naam zoeken

```
student@ubuntu-server-2:~$ sleep 1000 &
[1] 1783
student@ubuntu-server-2:~$ sleep 2000 &
[2] 1784
student@ubuntu-server-2:~$ pgrep sleep
1783
1784
student@ubuntu-server-2:~$ pgrep sleep -l
1783 sleep
1784 sleep
student@ubuntu-server-2:~$ pgrep sleep -a
1783 sleep 1000
1784 sleep 2000
student@ubuntu-server-2:~$ ps -C sleep
  PID TTY          TIME CMD
 1783 pts/0    00:00:00 sleep
 1784 pts/0    00:00:00 sleep
student@ubuntu-server-2:~$
```

processname achter
processid tonen

het kan ook met het **ps**
commando met
-C <command>

Basic Process Management

top

ordent processen naargelang gebruik van CPU of andere properties.

```
top - 18:47:05 up 25 min, 1 user, load average: 0.00, 0.00, 0.00
Tasks: 226 total, 1 running, 225 sleeping, 0 stopped, 0 zombie
%Cpu(s): 0.0 us, 0.0 sy, 0.0 ni,100.0 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
MiB Mem : 3889.8 total, 3088.6 free, 304.9 used, 496.4 buff/cache
MiB Swap: 0.0 total, 0.0 free, 0.0 used. 3356.7 avail Mem
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
14	root	20	0	0	0	0	I	0.3	0.0	0:00.42	rcu_sched
1789	student	20	0	10644	4060	3196	R	0.3	0.1	0:00.01	top
1	root	20	0	167676	12948	8236	S	0.0	0.3	0:03.32	systemd
2	root	20	0	0	0	0	S	0.0	0.0	0:00.01	kthreadd
3	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	rcu_gp
4	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	rcu_par_gp
5	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	netns
7	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kworker/0:0H-event+
9	root	0	-20	0	0	0	I	0.0	0.0	0:00.04	kworker/0:1H-event+
10	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	mm_percpu_wq

h → help

q → quit

Basic Process Management

```
0[          0.0%]   Tasks: 31, 36 thr; 1 running
1[          0.0%]   Load average: 0.00 0.00 0.00
2[|         0.7%]   Uptime: 00:27:07
3[          0.0%]
Mem[|||||    306M/3.80G]
Swp[         0K/0K]
```

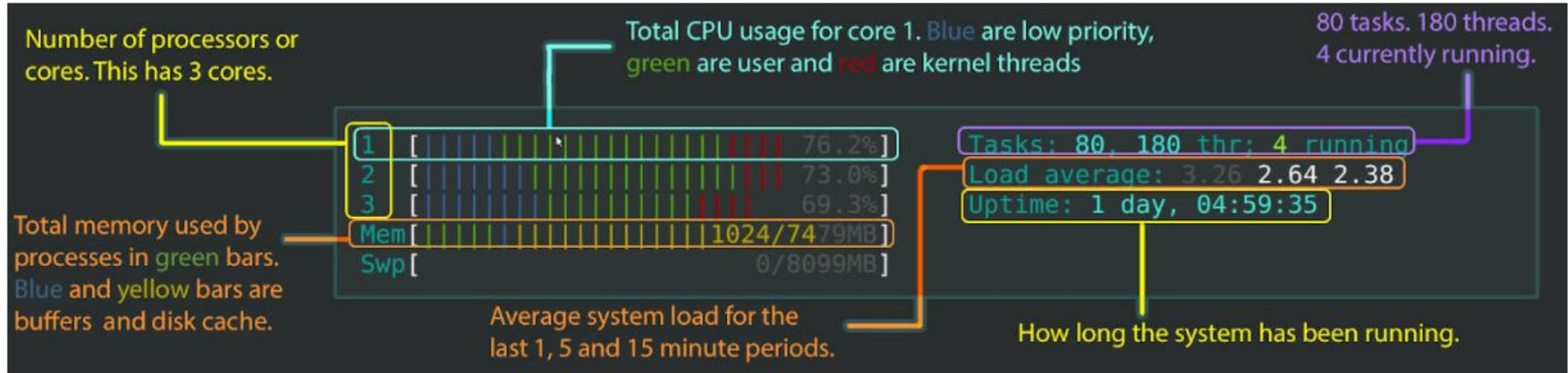
PID	USER	PRI	NI	VIRT	RES	SHR	S	CPU%	MEM%	TIME+	Command
1	root	20	0	163M	12948	8236	S	0.0	0.3	0:03.32	/sbin/init
540	root	19	-1	47896	15348	14248	S	0.0	0.4	0:00.22	/lib/systemd/systemd-journald
583	root	RT	0	346M	27364	9076	S	0.0	0.7	0:00.22	/sbin/multipathd -d -s
587	root	20	0	12384	7136	4440	S	0.0	0.2	0:00.32	/lib/systemd/systemd-udev
589	root	20	0	346M	27364	9076	S	0.0	0.7	0:00.00	/sbin/multipathd -d -s
590	root	RT	0	346M	27364	9076	S	0.0	0.7	0:00.00	/sbin/multipathd -d -s
591	root	RT	0	346M	27364	9076	S	0.0	0.7	0:00.00	/sbin/multipathd -d -s
592	root	RT	0	346M	27364	9076	S	0.0	0.7	0:00.00	/sbin/multipathd -d -s
593	root	RT	0	346M	27364	9076	S	0.0	0.7	0:00.10	/sbin/multipathd -d -s
594	root	RT	0	346M	27364	9076	S	0.0	0.7	0:00.00	/sbin/multipathd -d -s
788	systemd-t	20	0	89352	6544	5740	S	0.0	0.2	0:00.04	/lib/systemd/systemd-timesyncd
793	root	20	0	51124	11756	9748	S	0.0	0.3	0:00.03	/usr/bin/VGAAuthService
794	root	20	0	235M	9228	7768	S	0.0	0.2	0:01.63	/usr/bin/vmtoolsd
821	systemd-t	20	0	89352	6544	5740	S	0.0	0.2	0:00.00	/lib/systemd/systemd-timesyncd
822	root	20	0	235M	9228	7768	S	0.0	0.2	0:00.07	/usr/bin/vmtoolsd

F1Help F2Setup F3Search F4Filter F5Tree F6SortBy F7Nice - F8Nice + F9Kill F10Quit

Basic Process Management

htop

bovenste gedeelte



Basic Process Management

htop

onderste gedeelte

Niceness or the **user-space priority** of processes. Ranges from -20 (highest) to 19 (lowest.)

Process owner.

Process Ids.

Kernel-space priority of processes. Ranges from 0 to 139.

Total amount of virtual memory requested by processes. Not all may be in use.

Resident memory usage or what's currently being used by processes.

Shared memory used by processes.

Command that launched processes.

Processor time used by processes.

Process state:
R = runnable.
S = Interruptable sleep

Percentage of CPU time processes are currently using.

A task's current share of the physical memory. This is RES divided by total memory.


PID	USER	PRI	NI	VIRT	RES	SHR	S	CPU%	MEM%	TIME+	Command
17391	joe	20	0	1435M	285M	55844	S	42.6	3.8	55:25.38	simplescreenrecor
18081	joe	20	0	1435M	285M	55844	S	42.6	3.8	1:09.10	simplescreenrecor
18052	joe	20	0	2719M	100M	65044	S	41.9	1.3	0:52.05	geary
16157	root	20	0	310M	82448	45728	R	29.7	1.1	13:31.52	/usr/bin/X :0 -au
18065	joe	20	0	2719M	100M	65044	R	24.3	1.3	0:30.85	geary
18073	joe	30	10	1435M	285M	55844	S	24.3	3.8	0:31.82	simplescreenrecor
18074	joe	30	10	1435M	285M	55844	R	20.3	3.8	0:31.62	simplescreenrecor
18075	joe	30	10	1435M	285M	55844	S	20.3	3.8	0:31.44	simplescreenrecor
18077	joe	20	0	1435M	285M	55844	S	14.9	3.8	0:25.62	simplescreenrecor
18076	joe	20	0	1435M	285M	55844	S	8.8	3.8	0:14.99	simplescreenrecor
18082	joe	20	0	1435M	285M	55844	S	5.4	3.8	0:09.63	simplescreenrecor
16443	joe	9	13	505M	10480	7448	S	4.7	0.1	3:19.81	/usr/bin/pulseaud

Signalling Processes

kill

Process stoppen.

```
student@ubuntu-server-2:~$ sleep 1000 &
[1] 1885
student@ubuntu-server-2:~$ ps -f
UID          PID    PPID  C STIME TTY          TIME CMD
student      1354    1353  0 18:22 pts/0        00:00:00 -bash
student      1885    1354  0 18:50 pts/0        00:00:00 sleep 1000
student      1886    1354  0 18:50 pts/0        00:00:00 ps -f
student@ubuntu-server-2:~$ kill 1885
student@ubuntu-server-2:~$
[1]+  Terminated                  sleep 1000
student@ubuntu-server-2:~$ ps -f
UID          PID    PPID  C STIME TTY          TIME CMD
student      1354    1353  0 18:22 pts/0        00:00:00 -bash
student      1887    1354  0 18:50 pts/0        00:00:00 ps -f
student@ubuntu-server-2:~$
```



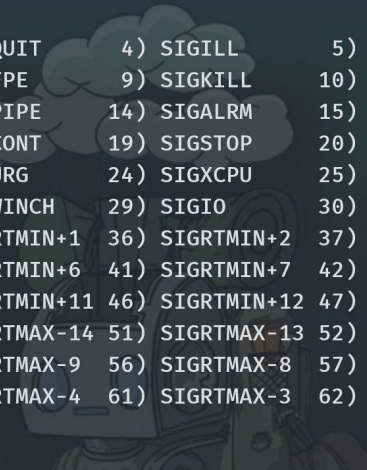
Signalling Processes

list signals

Draaiende processen kunnen signals ontvangen van andere processen of van users.

kill -l (letter L van list)

```
student@ubuntu-server-2:~$ kill -l
 1) SIGHUP      2) SIGINT      3) SIGQUIT     4) SIGILL      5) SIGTRAP
 6) SIGABRT     7) SIGBUS     8) SIGFPE     9) SIGKILL    10) SIGUSR1
11) SIGSEGV    12) SIGUSR2    13) SIGPIPE    14) SIGALRM    15) SIGTERM
16) SIGSTKFLT  17) SIGCHLD    18) SIGCONT    19) SIGSTOP    20) SIGTSTP
21) SIGTTIN    22) SIGTTOU    23) SIGURG     24) SIGXCPU    25) SIGXFSZ
26) SIGVTALRM  27) SIGPROF    28) SIGWINCH   29) SIGIO      30) SIGPWR
31) SIGSYS     34) SIGRTMIN   35) SIGRTMIN+1 36) SIGRTMIN+2 37) SIGRTMIN+3
38) SIGRTMIN+4 39) SIGRTMIN+5 40) SIGRTMIN+6 41) SIGRTMIN+7 42) SIGRTMIN+8
43) SIGRTMIN+9 44) SIGRTMIN+10 45) SIGRTMIN+11 46) SIGRTMIN+12 47) SIGRTMIN+13
48) SIGRTMIN+14 49) SIGRTMIN+15 50) SIGRTMAX-14 51) SIGRTMAX-13 52) SIGRTMAX-12
53) SIGRTMAX-11 54) SIGRTMAX-10 55) SIGRTMAX-9  56) SIGRTMAX-8  57) SIGRTMAX-7
58) SIGRTMAX-6 59) SIGRTMAX-5 60) SIGRTMAX-4 61) SIGRTMAX-3 62) SIGRTMAX-2
63) SIGRTMAX-1 64) SIGRTMAX
```



```
student@ubuntu-server-2:~$
```

Signalling Processes

kill -15

SIGTERM

standard kill

Wordt uitgevoerd als er geen signal wordt meegegeven.

Er wordt vriendelijk gevraagd aan het proces of het zich wil afsluiten. Het proces kan nog eerst een cleanup doen.

kill -9

SIGKILL

sure kill

Wordt niet naar het proces gestuurd, maar naar de kernel.

De kernel zal het proces stoppen! Er is geen kans tot cleanup.

Signalling Processes

pgrep

pgrep sleep

zou ook gaan, want
pgrep werkt met pattern
matching

killall

stuurt SIGTERM naar
alle processen met
de opgegeven naam
(geen pattern matching)

```
student@ubuntu-server-2:~$ pgrep sleep
2035
2036
student@ubuntu-server-2:~$ pkill sleep
[1]-  Terminated          sleep 1000
[2]+  Terminated          sleep 2000
student@ubuntu-server-2:~$ _
```

```
student@ubuntu-server-2:~$ sleep 1000 &
[1] 2044
student@ubuntu-server-2:~$ sleep 1500 &
[2] 2045
student@ubuntu-server-2:~$ jobs
[1]-  Running               sleep 1000 &
[2]+  Running               sleep 1500 &
student@ubuntu-server-2:~$ killall sleep
[1]-  Terminated          sleep 1000
[2]+  Terminated          sleep 1500
student@ubuntu-server-2:~$ _
```

Background Processes

jobs

jobs toont de jobs (processen) die in de background draaien in je huidige shell.

```
student@ubuntu-server-2:~$ jobs  
student@ubuntu-server-2:~$ _
```

Standaard draaien er geen jobs in de background.

Background Processes



Sommige processen kan je pauzeren (stoppen) naar de background met Ctrl-Z

→ SIGSTOP

```
student@ubuntu-server-2:~$ vim procdemo.txt  
  
[1]+  Stopped                  vim procdemo.txt  
student@ubuntu-server-2:~$ nano procdemo2.txt  
  
Use "fg" to return to nano.  
  
[2]+  Stopped                  nano procdemo2.txt  
student@ubuntu-server-2:~$
```

Background Processes

& (*ampersand*)

Processen die gestart worden met een **&** achteraan, worden naar de background gebracht waar ze blijven uitvoeren

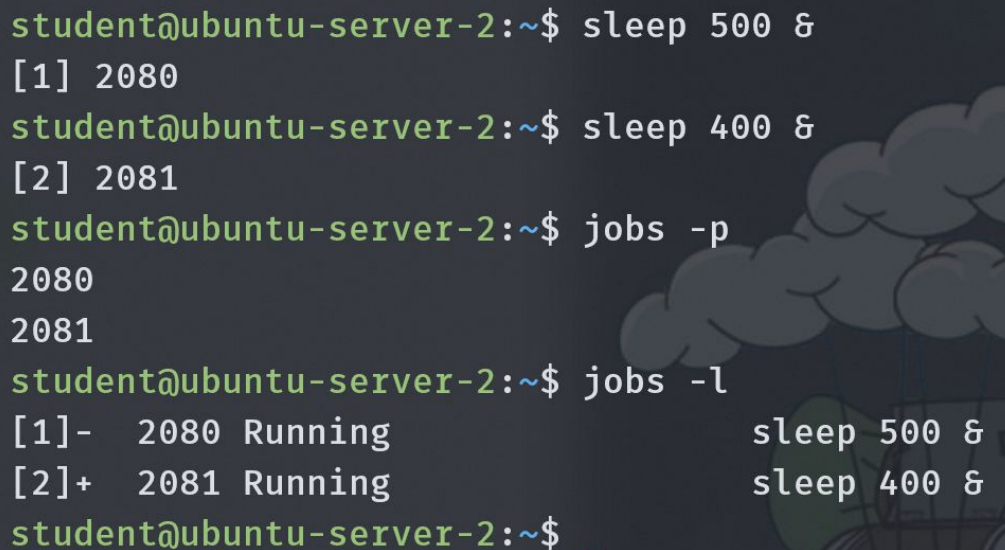
Ook deze jobs kunnen getoond worden met het commando `jobs`.

```
student@ubuntu-server-2:~$ sleep 1000 &
[3] 2070
student@ubuntu-server-2:~$ jobs
[1]-  Stopped                  vim procdemo.txt
[2]+  Stopped                  nano procdemo2.txt
[3]   Running                  sleep 1000 &
student@ubuntu-server-2:~$
```

Background Processes

`jobs -p` *of* `jobs -l`

Om de PID van de background processen te tonen.



```
student@ubuntu-server-2:~$ sleep 500 &
[1] 2080
student@ubuntu-server-2:~$ sleep 400 &
[2] 2081
student@ubuntu-server-2:~$ jobs -p
2080
2081
student@ubuntu-server-2:~$ jobs -l
[1]-  2080  Running                  sleep 500 &
[2]+  2081  Running                  sleep 400 &
student@ubuntu-server-2:~$
```

Background Processes

fg

Om een background proces naar de foreground te brengen en opnieuw door te laten uitvoeren.

De parameter is het nummer van de background job.

```
student@ubuntu-server-2:~$ jobs
[1]  Running                  sleep 1000 &
[2]-  Running                  sleep 2000 &
[3]+  Running                  sleep 3000 &
student@ubuntu-server-2:~$ fg 3
sleep 3000
```

fg voor een resume in foreground, van een suspended process kan uiteraard ook.

Background Processes

bg

Om een background proces dat gepauzeerd is terug te starten en op de background te houden.

```
student@ubuntu-server-2:~$ sleep 5000 &
[1] 2088
student@ubuntu-server-2:~$ sleep 3000
^Z
[2]+  Stopped                  sleep 3000
student@ubuntu-server-2:~$ jobs
[1]-  Running                  sleep 5000 &
[2]+  Stopped                  sleep 3000
student@ubuntu-server-2:~$ bg 2
[2]+ sleep 3000 &
student@ubuntu-server-2:~$ jobs
[1]-  Running                  sleep 5000 &
[2]+  Running                  sleep 3000 &
student@ubuntu-server-2:~$
```

Background Processes

SIGSTOP (19) en SIGCONT (18)

Om een background proces dat runt op de achtergrond opnieuw te pauzeren.

```
student@ubuntu-server-2:~$ sleep 1000 &
[1] 1896
student@ubuntu-server-2:~$ jobs
[1]+  Running                  sleep 1000 &
student@ubuntu-server-2:~$ kill -19 1896
student@ubuntu-server-2:~$ jobs
[1]+  Stopped                  sleep 1000
student@ubuntu-server-2:~$ kill -18 1896
student@ubuntu-server-2:~$ jobs
[1]+  Running                  sleep 1000 &
student@ubuntu-server-2:~$
```

of bg 1

Background Processes

+ (*plus*) en - (*minus*) bij **jobs**

Plusteken betekent het laatste proces dat naar de achtergrond is gebracht en is het default proces

→ als men geen jobnr opgeeft, zal het default proces aangesproken worden

Minteken betekent het voorlaatste proces dat naar de achtergrond is gebracht en krijgt het plusteken als het huidig default-proces (+) eindigt.

%- duidt het proces aan met het minteken

```
student@ubuntu-server-2:~$ sleep 1000 &
[1] 1889
student@ubuntu-server-2:~$ sleep 2000 &
[2] 1890
student@ubuntu-server-2:~$ sleep 3000 &
[3] 1891
student@ubuntu-server-2:~$ jobs
[1]  Running                sleep 1000 &
[2]-  Running                sleep 2000 &
[3]+  Running                sleep 3000 &
student@ubuntu-server-2:~$ fg
sleep 3000
^Z
[3]+  Stopped                sleep 3000
student@ubuntu-server-2:~$ jobs
[1]  Running                sleep 1000 &
[2]-  Running                sleep 2000 &
[3]+  Stopped                sleep 3000
student@ubuntu-server-2:~$ fg %-
sleep 2000
^Z
[2]+  Stopped                sleep 2000
student@ubuntu-server-2:~$
```

