

## Лабораторная работа. Управление процессами.

### Задачи

1. Вывод информации о процессах
2. Управление процессами

### Описание

Ниже приведены команды для работы в ОС Linux в консольном режиме. Изучите приведенные команды и отработайте их в командной строке Linux. При выполнении работы проявите творчество и поэкспериментируйте с командами.

### Управление процессами

Для демонстрации управления процессами будем использовать команду `ping`.

```
username@linux-pc:~$ ping localhost > /dev/null
```

Вывод `ping` будет перенаправляться в файл `/dev/null`, который представляет собой «чёрную дыру».

Обратите внимание, что терминал, похоже, «подвисает» на этой команде. Это происходит из-за выполнения команды на «переднем плане». Система будет продолжать выполнять `ping` до тех пор, пока процесс не будет прерван или приостановлен пользователем.

Прервите выполнение, нажав `Ctrl+C`

Затем для запуска того же процесса в фоне введите:

```
ping localhost > /dev/null &
```

Символ `&` запустит выполнение команды в фоновом режиме.

Обратите внимание, что предыдущая команда возвращает информацию об идентификаторе процесса:

```
[1] 158
```

Это означает, что этот процесс имеет номер задания **1** и идентификатор процесса (PID) **158**. Каждый терминал / оболочка будет иметь уникальные номера заданий. PID является общесистемным; Каждый процесс имеет уникальный идентификационный номер.

Чтобы узнать, какие команды запущены в текущем терминале, введите команду `jobs`.

```
username@linux-pc:~$ jobs
```

Вы можете перевести команду в интерактивный режим с помощью команды `fg`.

```
username@linux-pc:~$ fg %1
```

Обратите внимание, что команда `ping` снова управляет терминалом. Чтобы приостановить процесс и восстановить управление терминалом, введите `Ctrl-Z`.

Для продолжения выполнения процесса в фоне выполните команду `bg`.

```
username@linux-pc:~$ bg %1
```

Запустите еще несколько команд в фоне.

```
username@linux-pc:~$ ping localhost > /dev/null &
```

```
username@linux-pc:~$ ping localhost > /dev/null &
```

```
username@linux-pc:~$ ping localhost > /dev/null &
```

```
username@linux-pc:~$ jobs
```

Теперь, используя идентификатор процесса, остановите выполнение команд с помощью программы `kill`.

```
username@linux-pc:~$ kill %3
```

```
username@linux-pc:~$ jobs
```

Вы можете остановить все команды `ping` командой `killall`. После выполнения команды `killall` подождите несколько мгновений, а затем запустите команду `jobs`, чтобы проверить, что все процессы остановлены.

```
username@linux-pc:~$ killall ping
```

```
username@linux-pc:~$ jobs
```

Запустите несколько команд в фоне:

```
username@linux-pc:~$ ping localhost > /dev/null &
```

```
username@linux-pc:~$ ping localhost > /dev/null &
```

Выполните команду `top` для просмотра списка активных процессов.

```
username@linux-pc:~$ top
```

Команда `top` - это интерактивная программа, которая означает, что вы можете выдавать команды внутри программы. Вы будете использовать команду `top`, чтобы завершить процессы `ping`.

Введите букву `k`. Обратите внимание, что под `Swap` появилось приглашение:

```
top - 01:52:14 up 9:13, 1 user, load average: 0.04, 0.11, 0.19
Tasks: 10 total, 1 running, 9 sleeping, 0 stopped, 0 zombie
Cpu(s): 4.2%us, 2.6%sy, 0.0%ni, 93.0%id, 0.1%wa, 0.0%hi, 0.1%si, 0.0%st
Mem: 65969788k total, 8137188k used, 57832600k free, 286512k buffers
Swap: 2097148k total, 0k used, 2097148k free, 1110844k cached
```

**PID to kill:**

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
1	root	20	0	17868	2872	2624	S	0	0.0	0:00.14	init
19	syslog	20	0	171m	2796	2420	S	0	0.0	0:00.07	rsyslogd
23	root	20	0	19120	2020	1824	S	0	0.0	0:00.00	cron
25	root	20	0	50048	3416	2812	S	0	0.0	0:00.00	sshd
41	bind	20	0	376m	19m	5988	S	0	0.0	0:00.07	named
50	root	20	0	54460	2584	2176	S	0	0.0	0:00.00	login
62	sysadmin	20	0	18084	3200	2700	S	0	0.0	0:00.01	bash
103	sysadmin	20	0	6504	1824	1688	S	0	0.0	0:00.96	ping
104	sysadmin	20	0	6504	1788	1656	S	0	0.0	0:00.85	ping
108	sysadmin	20	0	17212	2240	2008	R	0	0.0	0:01.03	top

В приглашение PID to kill введите PID процесса для завершения и нажмите Enter.

```
top - 02:03:27 up 9:24, 1 user, load average: 0.53, 0.27, 0.23
Tasks: 10 total, 1 running, 9 sleeping, 0 stopped, 0 zombie
Cpu(s): 2.0%us, 1.8%sy, 0.0%ni, 95.9%id, 0.0%wa, 0.0%hi, 0.3%si, 0.0%st
Mem: 65969788k total, 7445364k used, 58524424k free, 288612k buffers
Swap: 2097148k total, 0k used, 2097148k free, 1096724k cached
Kill PID 112 with signal [15]:
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
114	sysadmin	20	0	17212	2192	1960	R	1	0.0	0:00.10	top
1	root	20	0	17868	2872	2624	S	0	0.0	0:00.14	init
19	syslog	20	0	171m	2796	2420	S	0	0.0	0:00.07	rsyslogd
23	root	20	0	19120	2020	1824	S	0	0.0	0:00.00	cron
25	root	20	0	50048	3416	2812	S	0	0.0	0:00.00	sshd
41	bind	20	0	376m	19m	5988	S	0	0.0	0:00.07	named
50	root	20	0	54460	2584	2176	S	0	0.0	0:00.00	login
62	sysadmin	20	0	18084	3200	2700	S	0	0.0	0:00.01	bash
112	sysadmin	20	0	6504	1844	1712	S	0	0.0	0:00.05	ping
113	sysadmin	20	0	6504	1812	1676	S	0	0.0	0:00.05	ping

Для отправки сигнала нажмите Enter еще раз.

Существует несколько различных числовых значений, которые могут быть отправлены процессу. Это предопределенные значения, каждый из которых имеет свое значение. Если вы хотите узнать больше об этих значениях, введите `man kill` в окне терминала.

В подсказке указано, что сигнал по умолчанию является сигналом завершения, обозначенным SIGTERM или номером 15.

Завершите другой процесс ping используя значение 9.

Сигнал kill 9 или SIGKILL является «сильным» сигналом, который нельзя игнорировать, в отличие от значения по умолчанию 15. Обратите внимание, что все ссылки на команду ping теперь удаляются из top.

```
[1]-  Terminated          ping localhost > /dev/null  
[2]+  Killed              ping localhost > /dev/null
```

Используйте q для завершения top и просмотра информации о завершенных процессах.

Завершение программ командами kill и pkill.

Запустите программы в фоне.

```
username@linux-pc:~$ sleep 888888 &
```

```
username@linux-pc:~$ sleep 888888 &
```

Команда sleep обычно используется для приостановки программы (сценария оболочки) в течение определенного периода времени. В этом случае он используется только для предоставления команды, которая займет много времени.

Используйте команду kill для завершения процесса sleep.

```
username@linux-pc:~$ ps
```

```
username@linux-pc:~$ kill <PID>
```

```
username@linux-pc:~$ jobs
```

Затем используйте pkill для завершения процесс, используя имя вместо PID.

```
username@linux-pc:~$ pkill -15 sleep
```

Команда ps по умолчанию выводит список процессов, запущенных в данном терминале.

```
username@linux-pc:~$ ping localhost > /dev/null &
```

```
username@linux-pc:~$ ps
```

Для вывода полного списка процессов используется опция -e (-A).

```
username@linux-pc:~$ ps -e
```