

AIM: Adam is working in an IT company. He has been given a task to reduce the load of a system by killing some of the processes running in the LINUX operating system. Which commands will he use to complete the given task with the help of the following operation?

- Kill processes by name

```
Feb 2 5:13 PM
m309@m309-BY-OEM:~$ ps aux
USER      PID %CPU %MEM    VSZ   RSS TTY      STAT START  TIME COMMAND
root        1  0.1  0.1 23736 14604 ?
root        2  0.0  0.0     0     0 ?          Ss  15:44  0:06 /sbin/init splash
root        3  0.0  0.0     0     0 ?          Ss  15:44  0:00 [kthreadd]
root        4  0.0  0.0     0     0 ?          Ss  15:44  0:00 [pool_workqueue_release]
root        5  0.0  0.0     0     0 ?          I<  15:44  0:00 [kworker/R-rcu_gp]
root        6  0.0  0.0     0     0 ?          I<  15:44  0:00 [kworker/R-kvfree_rcu_reclaim]
root        7  0.0  0.0     0     0 ?          I<  15:44  0:00 [kworker/R-slub_flushwq]
root        8  0.0  0.0     0     0 ?          I<  15:44  0:00 [kworker/R-netns]
root       11  0.0  0.0     0     0 ?          I<  15:44  0:00 [kworker/0:0H-events_highpri]
root       13  0.0  0.0     0     0 ?          I<  15:44  0:00 [kworker/R-mm_percpu_wq]
root       14  0.0  0.0     0     0 ?          I  15:44  0:00 [rcu_tasks_kthread]
root       15  0.0  0.0     0     0 ?          I  15:44  0:00 [rcu_tasks_rude_kthread]
root       16  0.0  0.0     0     0 ?          I  15:44  0:00 [rcu_tasks_trace_kthread]
root       17  0.0  0.0     0     0 ?          S  15:44  0:00 [ksoftirqd/0]
root       18  0.0  0.0     0     0 ?          I  15:44  0:00 [rcu_preempt]
root       19  0.0  0.0     0     0 ?          S  15:44  0:00 [rcu_exp_par_gp_kthread_worker/0]
root       20  0.0  0.0     0     0 ?          S  15:44  0:00 [rcu_exp_gp_kthread_worker]
root       21  0.0  0.0     0     0 ?          S  15:44  0:00 [migration/0]
root       22  0.0  0.0     0     0 ?          S  15:44  0:00 [idle_inject/0]
root       23  0.0  0.0     0     0 ?          S  15:44  0:00 [cpuhp/0]
root       24  0.0  0.0     0     0 ?          S  15:44  0:00 [cpuhp/1]
root       25  0.0  0.0     0     0 ?          S  15:44  0:00 [idle_inject/1]
root       26  0.0  0.0     0     0 ?          S  15:44  0:00 [migration/1]
root       27  0.0  0.0     0     0 ?          S  15:44  0:00 [ksoftirqd/1]
root       29  0.0  0.0     0     0 ?          I<  15:44  0:00 [kworker/1:0H-events_highpri]
root       30  0.0  0.0     0     0 ?          S  15:44  0:00 [cpuhp/2]
root       31  0.0  0.0     0     0 ?          S  15:44  0:00 [idle_inject/2]
root       32  0.0  0.0     0     0 ?          S  15:44  0:00 [migration/2]
root       33  0.0  0.0     0     0 ?          S  15:44  0:00 [ksoftirqd/2]
```

```
m309@m309-BY-OEM:~$ ps aux | grep firefox
root      8550  0.0  0.0     0     0 ?          I<  17:12  0:00 [kworker/u17:0]
m309      8557  1.7  0.6 698712 56156 ?
m309      8572  0.0  0.0 11024 4976 pts/0   Ss  17:12  0:00 bash
m309      8672  0.0  0.0 13748 4592 pts/0   R+  17:12  0:00 ps aux
m309@m309-BY-OEM:~$ ps aux | grep firefox
m309      8674  0.0  0.0  9144 2248 pts/0   S+  17:12  0:00 grep --color=auto firefox
m309@m309-BY-OEM:~$ kill 8674
bash: kill: (8674) - No such process
m309@m309-BY-OEM:~$
```

- Kill a process based on the process name

```
Feb 2 5:15 PM
m309@m309-BY-OEM:~ ps aux | grep firefox
m309@m309-BY-OEM:~$ kill 8674
bash: kill: (8674) - No such process
m309@m309-BY-OEM:~$ ps aux
USER      PID %CPU %MEM    VSZ   RSS TTY      STAT START  TIME COMMAND
root      8672  0.0  0.0  13748  4592 pts/0      R+  17:12  0:00 ps aux
m309@m309-BY-OEM:~$ ps aux | grep firefox
m309      8674  0.0  0.0  9144  2248 pts/0      S+  17:12  0:00 grep --color=auto firefox
m309@m309-BY-OEM:~$ kill 8674
bash: kill: (8674) - No such process
m309@m309-BY-OEM:~$ ps aux
```

```
Feb 2 5:16 PM
m309@m309-BY-OEM:~ ps aux | grep firebox
m309@m309-BY-OEM:~$ killall firebox
firebox: no process found
m309@m309-BY-OEM:~$
```

- Kill a single process at a time with the given process ID

```

Feb 2 5:18 PM m309@m309-BY-OEM:~ 
m309      9025 100 0.0 13748 4584 pts/0   R+  17:18  0:00 ps aux
m309@m309-BY-OEM:~ $ ps aux | grep chrome
m309      8072 11.4 3.1 34288156 255108 ?    Sl  17:11  0:51 /snap/chromium/3010/usr/lib/chromium-browser/chrome --password-store
=basic --disable-features=TFLiteLanguageDetectionEnabled
m309      8154 0.0 0.0 33568844 3592 ?    Sl  17:11  0:00 /snap/chromium/3010/usr/lib/chromium-browser/chrome_crashpad_handler
--monitor-self --monitor-self-annotation=ptype=crashpad-handler --database=/home/m309/snap/chromium/common/chromium/Crash Reports --an
notation=channel=snap --annotation=lsb-release=Ubuntu Core 22 --annotation=plat=Linux --annotation=prod=Chrome_Linux --annotation=ver=1
31.0.6778.139 --initial-client-fd=5 --shared-client-connection
m309      8156 0.0 0.0 33560632 3376 ?    S   17:11  0:00 /snap/chromium/3010/usr/lib/chromium-browser/chrome_crashpad_handler
--no-periodic-tasks --monitor-self-annotation=ptype=crashpad-handler --database=/home/m309/snap/chromium/common/chromium/Crash Reports
--annotation=channel=snap --annotation=lsb-release=Ubuntu Core 22 --annotation=plat=Linux --annotation=prod=Chrome_Linux --annotation=
ver=131.0.6778.139 --initial-client-fd=4 --shared-client-connection
m309      8160 0.0 0.7 33926264 60872 ?    S   17:11  0:00 /snap/chromium/3010/usr/lib/chromium-browser/chrome --type=zygote --
no-zygote-sandbox --string-annotations --crashpad-handler-pid=8154 --enable-crash-reporter=snap --change-stack-guard-on-fork=enable
m309      8161 0.0 0.7 33926252 60688 ?    S   17:11  0:00 /snap/chromium/3010/usr/lib/chromium-browser/chrome --type=zygote --
string-annotations --crashpad-handler-pid=8154 --enable-crash-reporter=snap --change-stack-guard-on-fork=enable
m309      8163 0.0 0.1 33926280 15620 ?    S   17:11  0:00 /snap/chromium/3010/usr/lib/chromium-browser/chrome --type=zygote --
string-annotations --crashpad-handler-pid=8154 --enable-crash-reporter=snap --change-stack-guard-on-fork=enable
m309      8271 2.1 2.3 34427040 185656 ?    Sl  17:11  0:09 /snap/chromium/3010/usr/lib/chromium-browser/chrome --type=gpu-proce
ss --string-annotations --crashpad-handler-pid=8154 --enable-crash-reporter=snap --change-stack-guard-on-fork=enable --gpu-preferences
=AAAAAAAAAAgAAAEAAAAAAAABgAAEAAAAAAAQAAAABAAAAAAAEEAAAAAAIAAAAAAAAgAAAAAAA --sha
red-files --field-trial-handle=3,i,14902952784849514346,9959533870011433259,262144 --disable-features=EyeDropper,TFLiteLanguageDetectio
nEnabled --variations-seed-version
m309      8288 0.7 1.3 33913600 110924 ?    Sl  17:11  0:03 /snap/chromium/3010/usr/lib/chromium-browser/chrome --type=utility -
-utility-sub-type=network.mojom.NetworkService --lang=en-US --service-sandbox-type=none --string-annotations --crashpad-handler-pid=815
4 --enable-crash-reporter=snap --change-stack-guard-on-fork=enable --shared-files=v8_context_snapshot_data:100 --field-trial-handle=3,
i,14902952784849514346,9959533870011433259,262144 --disable-features=EyeDropper,TFLiteLanguageDetectionEnabled --variations-seed-versio
n
m309      8298 0.0 0.6 33976752 53120 ?    Sl  17:11  0:00 /snap/chromium/3010/usr/lib/chromium-browser/chrome --type=utility -
-utility-sub-type=storage.mojom.StorageService --lang=en-US --service-sandbox-type=utility --string-annotations --crashpad-handler-pid=
8154 --enable-crash-reporter=snap --change-stack-guard-on-fork=enable --shared-files=v8_context_snapshot_data:100 --field-trial-handle
--string-annotations --crashpad-handler-pid=8154 --enable-crash-reporter=snap --change-stack-guard-on-fork=enable --lang=en-US --num-r
aster-threads=2 --enable-main-frame-before-activation --renderer-client-id=10 --time-ticks-at-unix-epoch=-1770027278097276 --launch-tim
e-ticks=5189675089 --shared-files=v8_context_snapshot_data:100 --field-trial-handle=3,i,14902952784849514346,9959533870011433259,262144
--disable-features=EyeDropper,TFLiteLanguageDetectionEnabled --variations-seed-version
m309      8477 27.6 5.5 1215597516 446052 ?    Sl  17:11  2:00 /snap/chromium/3010/usr/lib/chromium-browser/chrome --type=renderer
--string-annotations --crashpad-handler-pid=8154 --enable-crash-reporter=snap --change-stack-guard-on-fork=enable --lang=en-US --num-r
aster-threads=2 --enable-main-frame-before-activation --renderer-client-id=12 --time-ticks-at-unix-epoch=-1770027278097276 --launch-tim
e-ticks=5191883523 --shared-files=v8_context_snapshot_data:100 --field-trial-handle=3,i,14902952784849514346,9959533870011433259,262144
--disable-features=EyeDropper,TFLiteLanguageDetectionEnabled --variations-seed-version
m309      9012 0.0 0.8 1212078408 68036 ?    Sl  17:17  0:00 /snap/chromium/3010/usr/lib/chromium-browser/chrome --type=renderer
--string-annotations --crashpad-handler-pid=8154 --enable-crash-reporter=snap --change-stack-guard-on-fork=enable --lang=en-US --num-r
aster-threads=2 --enable-main-frame-before-activation --renderer-client-id=37 --time-ticks-at-unix-epoch=-1770027278097276 --launch-tim
e-ticks=5585528792 --shared-files=v8_context_snapshot_data:100 --field-trial-handle=3,i,14902952784849514346,9959533870011433259,262144
--disable-features=EyeDropper,TFLiteLanguageDetectionEnabled --variations-seed-version
m309      9044 0.0 0.0 9144 2140 pts/0   S+  17:18  0:00 grep --color=auto chrome
m309@m309-BY-OEM:~ $ kill 8156
m309@m309-BY-OEM:~ $ kill -9 8156
bash: kill: (8156) - No such process
m309@m309-BY-OEM:~ $

```

2. Write a program for process creation using C

- Orphan Process

The screenshot shows a terminal window titled "m309@m309-BY-OEM: ~". The window title bar includes icons for volume, brightness, and power. The terminal header displays "Feb 2 5:21 PM" and the file name "orphan.c *". The main area of the terminal contains the following C code:

```
GNU nano 7.2
#include <stdio.h>
#include <unistd.h>

int main() {
    int pid = fork();

    if (pid > 0) {
        // Parent process
        printf("Parent Process ID: %d\n", getpid());
        sleep(2);
    } else {
        // Child process
        sleep(5);
        printf("Child Process ID: %d\n", getpid());
        printf("New Parent Process ID: %d\n", getppid());
    }
    return 0;
}
```

At the bottom of the terminal window, there is a menu bar with keyboard shortcuts:

- ^G Help
- ^O Write Out
- ^W Where Is
- ^K Cut
- ^T Execute
- ^C Location
- M-U Undo
- M-A Set Mark
- ^X Exit
- ^R Read File
- ^V Replace
- ^U Paste
- ^J Justify
- ^/ Go To Line
- M-E Redo
- M-6 Copy

The screenshot shows a terminal window with the following session:

```
m309@m309-BY-OEM: ~$ nano orphan.c
m309@m309-BY-OEM: ~$ gcc orphan.c -o orphan
m309@m309-BY-OEM: ~$ ./orphan
Parent Process ID: 9285
m309@m309-BY-OEM: ~$ Child Process ID: 9286
New Parent Process ID: 2262
```

- Zombie Process

The screenshot shows a terminal window titled "zombie.c" with the following content:

```
GNU nano 7.2
#include <stdio.h>
#include <unistd.h>

int main() {
    int pid = fork();

    if (pid == 0) {
        // Child process
        printf("Child process exiting\n");
    } else {
        // Parent process
        sleep(10);
        printf("Parent process running\n");
    }
    return 0;
}
```

The terminal window has a dark background and light-colored text. The title bar says "zombie.c". The status bar at the top right shows "Feb 2 5:25 PM". The window has standard OS X-style controls (close, minimize, maximize) in the top right corner.

The screenshot shows a terminal window with the following session:

```
m309@m309-BY-OEM:~$ nano zombie.c
m309@m309-BY-OEM:~$ gcc zombie.c -o zombie
m309@m309-BY-OEM:~$ ./zombie
Child process exiting
Parent process running
m309@m309-BY-OEM:~$
```

The terminal window has a dark background and light-colored text. The prompt is "m309@m309-BY-OEM:~\$". The session shows the compilation of "zombie.c" into "zombie" and its execution, displaying the expected output of child and parent processes.

3. the process using fork () system call.

The screenshot shows a terminal window with two panes. The top pane displays the code for a C program named 'fork.c' using the nano text editor. The bottom pane shows the command-line history and output of the program execution.

```
GNU nano 7.2
#include <stdio.h>
#include <unistd.h>

int main() {
    pid_t pid = fork();

    if (pid == 0) {
        printf("Child Process\n");
        printf("Child PID: %d\n", getpid());
        printf("Parent PID: %d\n", getppid());
    } else {
        printf("Parent Process\n");
        printf("Parent PID: %d\n", getpid());
    }
    return 0;
}
```

File menu icon

Feb 2 5:26 PM

m309@m309-BY-OEM: ~

fork.c *

Q E

^G Help ^O Write Out ^W Where Is ^K Cut ^T Execute ^C Location M-U Undo M-A Set Mark
^X Exit ^R Read File ^\ Replace ^U Paste ^J Justify ^/ Go To Line M-E Redo M-6 Copy

```
m309@m309-BY-OEM:~$  
m309@m309-BY-OEM:~$ nano fork.c  
m309@m309-BY-OEM:~$ gcc fork.c -o fork  
m309@m309-BY-OEM:~$ ./fork  
Parent Process  
Parent PID: 9727  
Child Process  
Child PID: 9728  
Parent PID: 9727  
m309@m309-BY-OEM:~$
```

- Infinite loop on fork

Feb 2 5:30 PM

m309@m309-BY-OEM: ~

GNU nano 7.2

loop.c

```
#include <stdio.h>

int main() {
    while (1) {
        printf("Infinite loop running...\n");
    }
    return 0;
}
```

[Read 8 lines]

^G Help ^O Write Out ^W Where Is ^K Cut ^T Execute ^C Location M-U Undo M-A Set Mark
^X Exit ^R Read File ^\ Replace ^U Paste ^J Justify ^/ Go To Line M-E Redo M-6 Copy