



Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ: ИНФОРМАТИКА И СИСТЕМЫ УПРАВЛЕНИЯ

КАФЕДРА: КОМПЬЮТЕРНЫЕ СИСТЕМЫ И СЕТИ

НАПРАВЛЕНИЕ ПОДГОТОВКИ: 09.03.01 Информатика и вычислительная техника

О Т Ч Е Т

по лабораторной работе № 4

Тема: Процессы

Дисциплина: Операционные системы

Студент

ИУ6-52Б
(Группа)

16.09.24
(Подпись, дата)

А. П. Плютто
(И. О. Фамилия)

Преподаватель

16.09.24
(Подпись, дата) В.Ю. Мельников
(И. О. Фамилия)

Москва, 2024

Содержание

1. Введение	3
1.1. Цель работы	3
1.2. Задание	3
2. Выполнение	4
2.1. Открыть в текстовом браузере некую страницу и перевести его в фоновый режим	4
2.2. Запустить ещё два экземпляра текстового браузера	4
2.3. Найти процесс, максимально нагружающий процессор	5
2.4. Вывести список процессов текущего пользователя	6
2.5. «Убить» первый процесс браузера в котором открыта 1 страница	7
2.6. Вывести список всех процессов всех пользователей	7
2.7. Просмотреть список процессов постранично	8
2.8. Отобрать из вывода команды <code>ps</code> строку, соответствующую процессу <code>dbus-daemon</code> , определить, где лежит её выполняемый файл и с какими параметрами он запущен	8
2.9. Записать в файл с именем, содержащим текущее время, строку «-----» и список процессов	9
2.10. Выполнить команду в фоновом режиме с отсрочкой запуска на 1 минуту. Продемонстрировать, что команда выполнилась именно через минуту.	10
2.11. Отобрать из одного из сформированных файлов строки, относящиеся к одному из процессов.	11
2.12. Вывести результаты работы произвольной команды в один файл, а сообщения об ошибках в другой. Продемонстрировать правильность работы. .	12
2.13. Выполнить произвольную команду с ограничением использования процессорного времени 300 секунд и выводом результатов и сообщений об ошибках в один файл.	13
2.14. Настроить <code>cron</code> на выполнение команды ежедневно в заданное время. Продемонстрировать правильность работы.	13

1. Введение

1.1. Цель работы

Получение теоретических и практических сведений об управлении процессами, потоками и оперативной памятью в UNIX-подобных системах и в Linux в частности.

1.2. Задание

- Открыть в текстовом браузере некую страницу и перевести его в фоновый режим
- Запустить ещё два экземпляра текстового браузера в фоновом режиме
- Найти процесс, максимально нагружающий процессор
- Вывести список процессов текущего пользователя
- «Убить» первый процесс браузера в котором открыта 1 страница
- Вывести список всех процессов всех пользователей
- Просмотреть список процессов постранично
- Отобрать из вывода команды ps строку, соответствующую процессу «dbus-daemon», определить, где лежит её выполняемый файл и с какими параметрами он запущен
- Записать в файл с именем, содержащим текущее время, строку «-----» и список процессов
- Выполнить команду в фоновом режиме с отсрочкой запуска на 1 минуту. Продемонстрировать, что команда выполнилась именно через минуту.
- Отобрать из одного из сформированных файлов строки, относящиеся к одному из процессов.
- Вывести результаты работы произвольной команды в один файл, а сообщения об ошибках в другой. Продемонстрировать правильность работы.
- Выполнить произвольную команду с ограничением использования процессорного времени 300 секунд и выводом результатов и сообщений об ошибках в один файл.
- Настроить cron на выполнение команды ежедневно в заданное время. Продемонстрировать правильность работы.

2. Выполнение

2.1. Открыть в текстовом браузере некую страницу и перевести его в фоновый режим

Откроем в браузере `w3m` страницу `google.com`

Используя сочетания клавиш `Ctrl+z` переведем страницу в фоновый режим.

2.2. Запустить ещё два экземпляра текстового браузера

Откроем страницу `ya.ru` и переведем ее в фоновый режим.

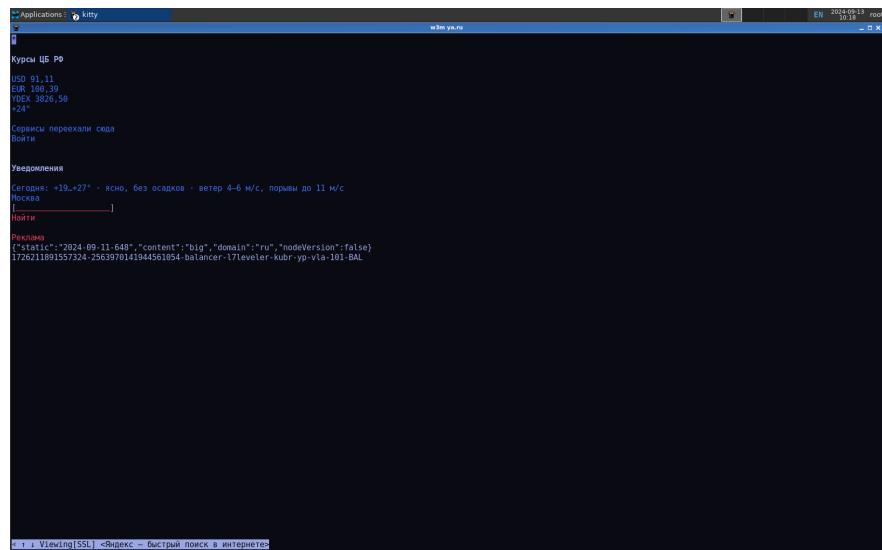


Рисунок 1 — `ya.ru`

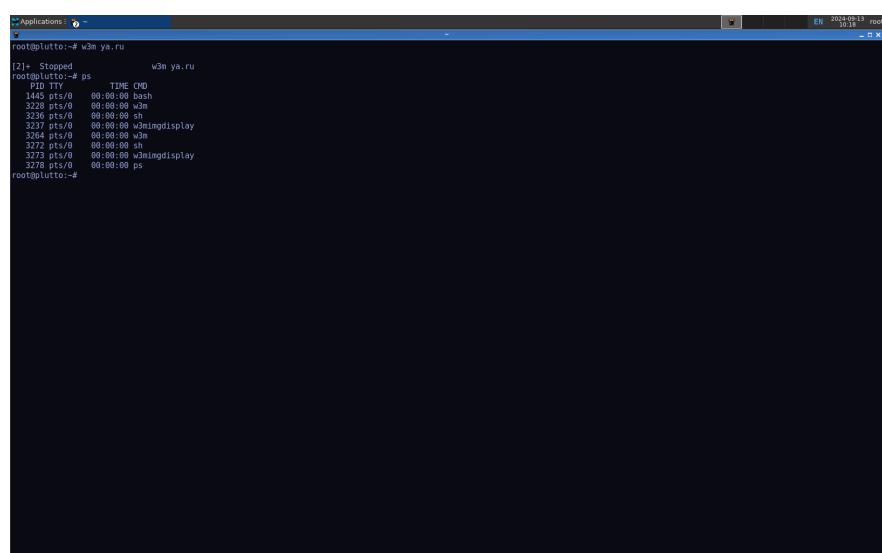


Рисунок 2 — Страница переведена в фоновый режим и отображается в списке процессов

Откроем страницу cppreference.com и переведем ее в фоновый режим.

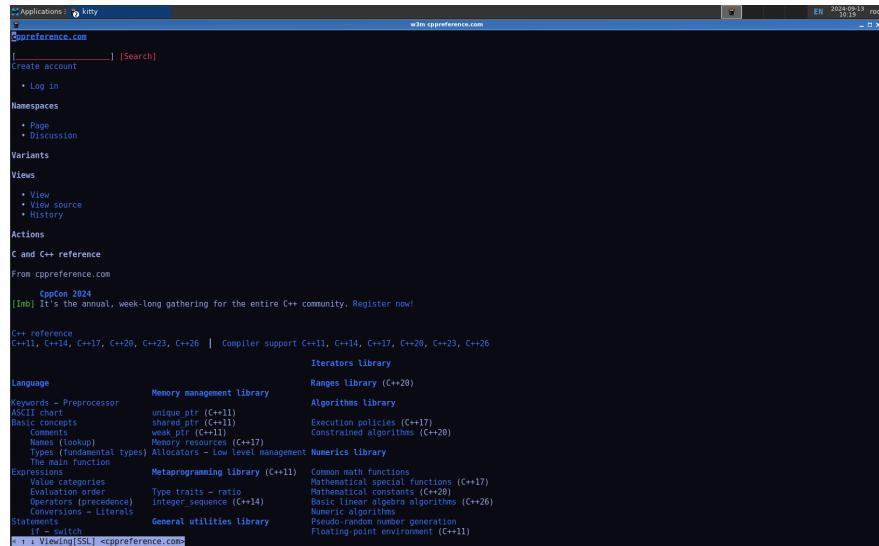


Рисунок 3 — cppreference.com

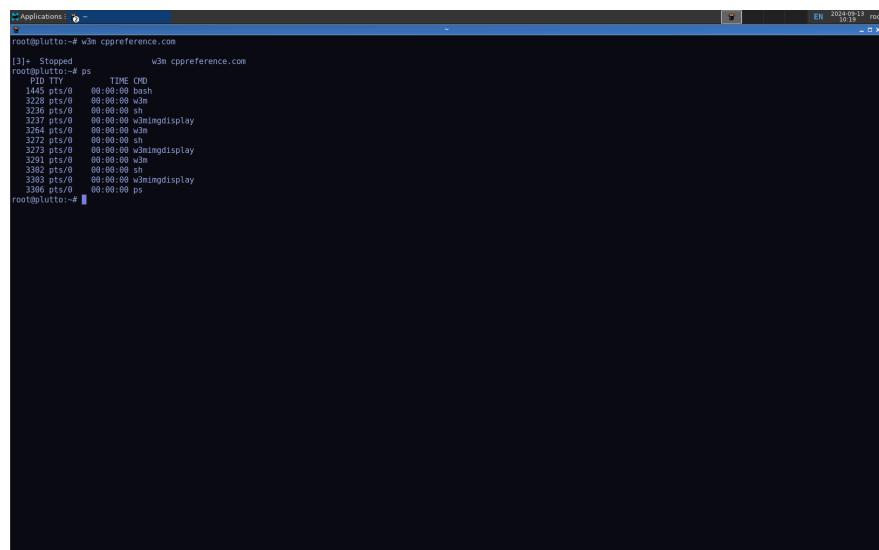


Рисунок 4 — Страница переведена в фоновый режим и отображается в списке процессов

2.3. Найти процесс, максимально нагружающий процессор

Для того, чтобы найти такой процесс выполним команду [top](#) :

```

Applications: 6 khttp
EN 2024-05-13 10:20:57 root

Top - 10:20:57 up 30 min, 4 users, load average: 0.01, 0.05, 0.08
Tasks: 192 total, 4 running, 185 sleeping, 3 stopped, 0 zombie
%Cpu(s): 1.0 us, 0.8 sy, 0.0 id, 97.1 jd, 0.0 wa, 0.0 hi, 0.3 sl, 0.0 st
Mem: 32.0G used, 18.0G free, 2.0G available, 540.0G cached
Swap: 376.0G total, 720.2G free, 255.8G used. 1422.4G avail Mem

20:00:00 19 -1 507912 69100 46648 R 2.0 3.4 0:30.18 Xorg
1349 root 20 0 613700 15540 13492 R 1.3 0.8 0:30.30 conky
1350 root 20 0 404616 99972 51984 S 0.7 4.5 0:02.05 kitty
1351 root 20 0 404616 100022 51984 S 0.7 4.5 0:02.05 kitty-registr
1734 root 20 0 466188 93636 51924 S 0.3 4.6 0:00.54 x-terminal-emul
3315 root 20 0 9724 4740 2692 R 0.3 0.2 0:00.08 top
root 20 0 23693 10286 6952 S 0.1 0.0 0:00.00 kworker/0:0-events_highpri
2 root 0 0 0 0 0 0 0 0 0:00.00 kworker/0:0-kmem_pcpu
2 root 0 0 0 0 0 0 0 0 0:00.00 kworker/0:0-kmem_pcpu
3 root 20 0 0 0 0 0 0 0 0:00.00 pool_workqueue_release
4 root 0 -29 0 0 0 0 0 0 0:00.00 kworker/R-rcu_0q
5 root 0 -29 0 0 0 0 0 0 0:00.00 kworker/R-rcu_qs
6 root 0 -29 0 0 0 0 0 0 0:00.00 kworker/R-rcu_qs_flushq
7 root 0 -29 0 0 0 0 0 0 0:00.00 kworker/R-netns
8 root 0 -29 0 0 0 0 0 0 0:00.00 kworker/R-poll_maintenance_highpri
12 root 0 -29 0 0 0 0 0 0 0:00.00 kworker/R-mm_percpu_wq
13 root 20 0 0 0 0 0 0 0 0:00.00 rCU_tasks_rude_kthread
14 root 20 0 0 0 0 0 0 0 0:00.00 rCU_tasks_trace_kthread
15 root 20 0 0 0 0 0 0 0 0:00.00 rCU_tasks_trace_rq
16 root 20 0 0 0 0 0 0 0 0:00.41 rCU_sched
17 root 20 0 0 0 0 0 0 0 0:00.00 rCU_exp_par_gp_kthread_worker/0
18 root 20 0 0 0 0 0 0 0 0:00.00 rCU_exp_gp_kthread_worker
19 root 0 -29 0 0 0 0 0 0 0:00.00 rCU_exp_gp_kthread_worker/0
20 root 0 0 0 0 0 0 0 0 0:00.00 cpuhp/0
21 root 20 0 0 0 0 0 0 0 0:00.00 cpuhp/0
22 root 0 -29 0 0 0 0 0 0 0:00.00 kworker/1:0h_events_highpri
23 root 20 0 0 0 0 0 0 0 0:00.00 ksftiotrq/0
25 root 0 -29 0 0 0 0 0 0 0:00.00 kworker/1:0h_events_highpri
27 root 20 0 0 0 0 0 0 0 0:00.00 kdevtmpfs
28 root 0 -29 0 0 0 0 0 0 0:00.00 kworker/0:0-inet_frag_wq
36 root 20 0 0 0 0 0 0 0 0:00.00 kauditd
31 root 20 0 0 0 0 0 0 0 0:00.00 khungtaskd
32 root 20 0 0 0 0 0 0 0 0:00.00 kworker/R-fan
33 root 0 -29 0 0 0 0 0 0 0:00.00 kworker/R-writeback
34 root 20 0 0 0 0 0 0 0 0:00.23 kcompactd0
35 root 25 0 0 0 0 0 0 0 0:00.00 ksm
36 root 39 19 0 0 0 0 0 0 0:00.68 kmemcgpagepd
37 root 0 -29 0 0 0 0 0 0 0:00.00 kworker/R-kintegrityrd
38 root 0 -29 0 0 0 0 0 0 0:00.00 kworker/R-kblockd
39 root 0 -29 0 0 0 0 0 0 0:00.00 kworker/R-kmemcg_reclaim_blo
41 root 0 -29 0 0 0 0 0 0 0:00.00 kworker/R-tcm-dev_wq
42 root 0 -29 0 0 0 0 0 0 0:00.00 kworker/R-edac-poller
43 root 0 -29 0 0 0 0 0 0 0:00.00 kworker/R-defreq_wq
44 root 0 -51 0 0 0 0 0 0 0:00.00 kworker/R-lll
45 root 0 -29 0 0 0 0 0 0 0:00.00 kworker/B-1H-kblockd
46 root 20 0 0 0 0 0 0 0:00.75 kswapd0

```

Рисунок 5 — [top](#)

Получается, что самым нагружающим процессом является `Xorg`.

2.4. Вывести список процессов текущего пользователя

Удобнее команды `top` есть команда `ps`, если ее использовать с аргументом `a`, то получится вывести список процессов текущего пользователя.

Рисунок 6 — ps a

```

root@Pluto:~# ps aux
  PID TTY      TIME CMD
 847 ttys1    S+  0:00 /usr/libexec/gsd-wacom
 850 ttys1    S+  0:00 /usr/libexec/gsd-color
 852 ttys1    S+  0:00 /usr/libexec/gsd-keyboard
 856 ttys1    S+  0:00 /usr/libexec/gsd-prism-notifications
 873 ttys1    S+  0:00 /usr/libexec/gsd-tile-wall
 876 ttys1    S+  0:00 /usr/libexec/gsd-smartcard
 889 ttys1    S+  0:00 /usr/libexec/gsd-datetime
 894 ttys1    S+  0:00 /usr/libexec/gsd-screensaver-proxy
 896 ttys1    S+  0:00 /usr/libexec/gsd-sound
 910 ttys1    S+  0:00 /usr/libexec/gsd-weather
 911 ttys1    S+  0:00 /usr/libexec/gsd-settings
 919 ttys1    S+  0:00 /usr/libexec/gsd-housekeeping
 928 ttys1    S+  0:00 /usr/libexec/gsd-power
 956 ttys1    S+  0:00 /usr/libexec/gsd-printer
1088 ttys1    S+  0:00 /usr/libexec/gsd-video-frames
1096 ttys1    S+  0:00 ibus-daemon - panel disable -r -xim
1097 ttys1    S+  0:00 /usr/bin/uis -m /usr/share/gnome-shell/org.gnome.ScreenSaver
1100 ttys1    S+  0:00 /usr/libexec/ibus-extension-gtk3
1106 ttys1    S+  0:01 /usr/libexec/ibus-x11 -kill-demon
1110 ttys1    S+  0:00 /usr/libexec/ibus-x11 -kill-demon
1111 ttys1    S+  0:00 /usr/libexec/ibus-engine-simple
1112 ttys1    S+  0:00 /usr/libexec/ibus-engine-simple
1157 ttys1    S+  0:00 /usr/libexec/dconf-service
1208 ttys2    S+  0:00 bash
1209 ttys2    S+  0:00 /bin/sh /usr/bin/startx
1210 ttys2    S+  0:00 xinit /etc/X11/xinit/xinirc -- /etc/X11/xinit/xserverrc :0 vt2 -keetty -auth /tmp/serverauth.Xs7gXIpOB
1220 ttys2    S+  0:31 /usr/lib/xorg/Xorg -nolisten tcp :0 vt2 -keetty -auth /tmp/serverauth.Xs7gXIpOB
1221 ttys2    S+  0:00 /bin/sh /usr/lib/arch64-linux-gnu/openbox-autostart openbox
1339 ttys2    S+  0:00 sh /etc/xdg/openbox/autostart
1340 ttys2    R+  0:00 xfce4-panel
1341 ttys2    S+  0:00 xfce4-panel
1340 ttys2    S+  0:00 xfce4-panel
1340 ttys2    S+  0:31 conky
1350 ttys2    S+  0:02 kwin_start -start- maximized
1443 ttys2    S+  0:00 /usr/lib/arch64-linux-gnu/xfce4/panel/wrapper-2.0 /usr/lib/arch64-linux-gnu/xfce4/panel/plugins/libsystray.so 6 14680076 systray Status Tray Plugin Provide
1445 pts/0    S+  0:00 /bin/bash -posix
1451 ttys2    S+  0:00 /usr/lib/arch64-linux-gnu/xfce4/panel/wrapper-2.0 /usr/lib/arch64-linux-gnu/xfce4/panel/plugins/libactions.so 14 14680077 actions Action Buttons Log out, l
1455 ttys2    S+  0:00 /usr/lib/arch64-linux-gnu/xfce4/panel/wrapper-2.0 /usr/lib/arch64-linux-gnu/xfce4/panel/plugins/libactions.so 8 14680087 actions Action Buttons Log out, l
1734 ttys2    S+  0:00 x-terminal-emulator
1737 pts/1    S+* 0:00 /bin/bash -posix
3226 pts/1    T+  0:00 wdm process
3230 pts/0    S+  0:00 /usr/lib/x3m/w3mimgdisplay
3231 pts/0    S+  0:00 /usr/lib/x3m/w3mimgdisplay
3264 pts/0    S+  0:00 wdm ya.ru
3271 pts/0    S+  0:00 /usr/lib/x3m/w3mimgdisplay
3277 pts/0    S+  0:00 /usr/lib/x3m/w3mimgdisplay
3291 pts/0    T+  0:00 wdm cpreference.com
3300 pts/0    S+  0:00 wdm ya.ru /usr/lib/x3m/w3mimgdisplay
3304 pts/0    S+  0:00 /usr/lib/x3m/w3mimgdisplay
3315 pts/0    T+  0:00 top
3347 pts/0    R+  0:00 ps a
root@Pluto:~#

```

Рисунок 7 — `ps a`

2.5. «Убить» первый процесс браузера в котором открыта 1 страница

Так как страница переведена в фоновый режим, то закрыть ее обычной командой `kill` передав аргументом `PID = 3228`, но можно добавить флаг `-9`, который означает принудительное завершение.

```

root@Pluto:~# ps
  PID TTY      TIME CMD
1445 pts/0    0:00:00 bash
3226 pts/0    0:00:00 sh
3228 pts/0    0:00:00 sh
3264 pts/0    0:00:00 wdm
3271 pts/0    0:00:00 sh
3277 pts/0    0:00:00 w3mimgdisplay
3291 pts/0    0:00:00 sh
3300 pts/0    0:00:00 sh
3302 pts/0    0:00:00 w3mimgdisplay
3315 pts/0    0:00:00 top
3344 pts/0    0:00:00 ps
[1]  killed          w3m google.com
root@Pluto:~# ps
  PID TTY      TIME CMD
1445 pts/0    0:00:00 bash
3264 pts/0    0:00:00 wdm
3271 pts/0    0:00:00 sh
3277 pts/0    0:00:00 w3mimgdisplay
3291 pts/0    0:00:00 sh
3302 pts/0    0:00:00 sh
3303 pts/0    0:00:00 w3mimgdisplay
3315 pts/0    0:00:00 top
3344 pts/0    0:00:00 ps
root@Pluto:~#

```

Рисунок 8 — `kill -9 3228`

Хоть и не сразу, но процесс завершился.

2.6. Вывести список всех процессов всех пользователей

Для того, чтобы вывести список всех процессов всех пользователей используем ту же команду `ps` с 3 аргументами: `aux`.

```

root      1 0.0  0.0 29392 1744 ? S 09:58 0:00 (sd-pam)
Debian+ 696 0.0  0.0 23932 3096 ? Ss 09:58 0:00 /usr/bin/pipewire
Debian+ 714 0.0  0.1 10272 3096 ? Ss 09:58 0:00 /usr/bin/pipewire -filter-chain.conf
debian+ 716 0.0  0.0 10272 6972 ? Ss 09:58 0:00 /usr/bin/pipewire-registrar
Debian+ 717 0.0  0.2 18984 3084 ? Ss 09:58 0:00 /usr/bin/pulseaudio-pulse
Debian+ 718 0.0  0.2 23436 5852 ttyl Ss+ 09:58 0:00 /usr/libexec/gdm-wayland-session --register-session dbus-run-session --gnome-session --autostart /usr/share/gdm/greeter
Debian+ 721 0.0  0.1 7860 2912 ? Ss 09:58 0:00 /usr/bin/dbus-daemon --session --address=systemd: --nofork --nopidfile --systemd-activation --syslog-only
Debian+ 724 0.0  0.0 10272 3096 ? Ss 09:58 0:00 /usr/bin/dbus-daemon --session --autostart /usr/share/gdm/greeter/autostart
rfkitd 728 0.0  0.1 20388 2324 ? Ss 09:58 0:00 /usr/libexec/rfkit-dæmon
Debian+ 729 0.0  0.1 10472 3024 ttyl Ss+ 09:58 0:00 dbus-daemon@/crtkit-dæmon -print-address -4 -session
Debian+ 730 0.0  0.1 10472 3024 ttyl Ss+ 09:58 0:00 dbus-daemon@/crtkit-dæmon -print-address -4 -session
Debian+ 757 0.1  0.5 367669 115764 ttyl Ss+ 09:58 0:00 /usr/libexec/gdm-session-binary --autostart /usr/share/gdm/greeter/autostart
Debian+ 786 0.0  0.2 38236 5996 ttyl Ss+ 09:58 0:00 /usr/libexec/at-spi-bus-launcher
Debian+ 792 0.0  0.1 10272 3096 ttyl Ss+ 09:58 0:00 /usr/bin/dbus-daemon --session --nofork --print-address 11 -address
Debian+ 794 0.0  0.9 32256 40292 ttyl Ss+ 09:58 0:00 /usr/bin/kmod -W 1000 -L 1000 -notset accessx -core -auth /run/user/108/.mutter-Xwayland.0KWTZ -listenfd 4
Debian+ 796 0.0  0.2 23594 6808 ttyl Ss+ 09:58 0:00 /usr/libexec/at-spi2-registryd -use-gnome-session
colorbar 802 0.0  0.0 31792 6448 ? Ss+ 09:58 0:00 /usr/libexec/gdm-wayland-session
root 835 0.0  0.3 318496 7544 ? Ss+ 09:58 0:00 /usr/libexec/gpword
Debian+ 837 0.0  0.8 260218 16684 ttyl Ss+ 09:58 0:00 /usr/bin/gjs -m /usr/share/gnome-shell/org.gnome.Shell.Notifications
Debian+ 839 0.0  0.0 10272 3096 ? Ss 09:58 0:00 /usr/bin/dbus-daemon --session --nofork --nopidfile --systemd-activation
Debian+ 843 0.0  0.6 412928 12156 ttyl Ss+ 09:58 0:00 /usr/libexec/gsd-wacom
Debian+ 858 0.0  0.6 414436 12984 ttyl Ss+ 09:58 0:00 /usr/libexec/gsd-color
Debian+ 853 0.0  0.6 485436 12484 ttyl Ss+ 09:58 0:00 /usr/libexec/gsd-keyboard
Debian+ 852 0.0  0.6 485436 12484 ttyl Ss+ 09:58 0:00 /usr/libexec/gsd-keyboard-notifications
Debian+ 873 0.0  0.2 53912 5296 ttyl Ss+ 09:58 0:00 /usr/libexec/gsd-rkill
Debian+ 876 0.0  0.2 459588 5972 ttyl Ss+ 09:58 0:00 /usr/libexec/gsd-smartcard
Debian+ 880 0.0  0.0 10272 3096 ? Ss 09:58 0:00 /usr/libexec/gsd-timer
Debian+ 892 0.0  0.7 747556 16812 ttyl Ss+ 09:58 0:00 /usr/libexec/gsd-netlink-keys
Debian+ 898 0.0  0.2 308720 5276 ttyl Ss+ 09:58 0:00 /usr/libexec/gsd-screensaver-proxy
Debian+ 904 0.0  0.0 10272 3096 ? Ss 09:58 0:00 /usr/libexec/gsd-screensaver
Debian+ 916 0.0  0.2 38316 4996 ttyl Ss+ 09:58 0:00 /usr/libexec/gsd-alloy-settings
Debian+ 919 0.0  0.3 458668 6792 ttyl Ss+ 09:58 0:00 /usr/libexec/gsd-housekeeping
Debian+ 928 0.0  0.7 525924 15284 ttyl Ss+ 09:58 0:00 /usr/libexec/gsd-power
Debian+ 934 0.0  0.0 10272 3096 ? Ss 09:58 0:00 /usr/libexec/gsd-power
Debian+ 1088 0.0  0.2 788788 54088 ttyl Ss+ 09:58 0:00 /usr/libexec/mutter-x11-frames
Debian+ 1099 0.0  0.4 461928 8272 ttyl Ss+ 09:58 0:00 ibus-daemon --panel disable -r -xim
Debian+ 1100 0.0  0.0 308928 13520 ttyl Ss+ 09:58 0:00 /usr/libexec/gdm-shutdown/gnome-shell.org.gnome.ScreenSaver
Debian+ 1105 0.0  0.3 389892 13520 ttyl Ss+ 09:58 0:00 /usr/libexec/gdm-doctor
Debian+ 1106 0.0  0.8 420952 17584 ttyl Ss+ 09:58 0:01 /usr/libexec/bus-extension-on-k3
Debian+ 1110 0.0  0.2 535792 36148 ttyl Ss+ 09:58 0:00 /usr/libexec/gdm-x11 -kill-demon
Debian+ 1111 0.0  0.3 253748 6240 ttyl Ss+ 09:58 0:00 /usr/libexec/gdm-x11-kill-demon
Debian+ 1113 0.0  0.3 235888 6240 ttyl Ss+ 09:58 0:00 /usr/libexec/gdm-engine-simple
root 1146 0.0  0.2 310232 5728 ? Ss 09:58 0:00 /usr/libexec/power-engine-daemon
Debian+ 1152 0.0  0.0 10272 3096 ? Ss 09:58 0:00 /usr/libexec/power-profile-service
root 1167 0.0  0.1 10184 2336 ? Ss+ 09:58 0:00 login - root
root 1188 0.0  0.3 21512 7496 ? Ss+ 09:58 0:00 /usr/lib/systemd/systemd -user
root 1190 0.0  0.0 10272 3096 ? Ss 09:58 0:00 /sbin/init
root 1195 0.0  0.0 8 0 ? Ss+ 09:58 0:00 /bin/sh
root 1206 0.0  0.0 5796 1588 ttyl Ss+ 09:58 0:00 /bin/bash
root 1208 0.0  0.0 2388 1296 ttyl Ss+ 09:58 0:00 /bin/sh /etc/X11/xserverrc .v12 _keepetty -auth /tmp/serverauth.X7oTp0Vb
root 1210 0.0  0.1 4584 2128 ttyl Ss+ 09:58 0:00 /init /etc/X11/xinit/xinitrc -- /etc/X11/xinit/xserverrc .v12 _keepetty -auth /tmp/serverauth.X7oTp0Vb

```

Рисунок 9 — `ps axu`

2.7. Просмотреть список процессов постранично

Воспользуемся командой `more` чтобы вывести тот же список, но постранично: `ps axu | more`

```

root      1 0.0  0.0 29392 1744 ? S 09:58 0:00 (sd-pam)
Debian+ 696 0.0  0.0 23932 3096 ? Ss 09:58 0:00 /usr/bin/pipewire
Debian+ 714 0.0  0.1 10272 3096 ? Ss 09:58 0:00 /usr/bin/pipewire -filter-chain.conf
debian+ 716 0.0  0.0 10272 3096 ? Ss 09:58 0:00 /usr/bin/pipewire-registrar
Debian+ 717 0.0  0.2 18984 3084 ? Ss 09:58 0:00 /usr/bin/pulseaudio-pulse
Debian+ 718 0.0  0.2 23436 5852 ttyl Ss+ 09:58 0:00 /usr/libexec/gdm-wayland-session --register-session dbus-run-session --gnome-session --autostart /usr/share/gdm/greeter
Debian+ 721 0.0  0.1 7860 2912 ? Ss 09:58 0:00 /usr/libexec/gdm-wayland-session
rfkitd 728 0.0  0.1 20388 2324 ? Ss 09:58 0:00 /usr/libexec/rfkit-dæmon
Debian+ 729 0.0  0.1 10472 3024 ttyl Ss+ 09:58 0:00 dbus-daemon@/crtkit-dæmon -print-address -4 -session
Debian+ 730 0.0  0.1 10472 3024 ttyl Ss+ 09:58 0:00 dbus-daemon@/crtkit-dæmon -print-address -4 -session
Debian+ 757 0.1  0.5 367669 115764 ttyl Ss+ 09:58 0:00 /usr/libexec/gdm-session-binary --autostart /usr/share/gdm/greeter/autostart
Debian+ 786 0.0  0.2 38236 5996 ttyl Ss+ 09:58 0:00 /usr/libexec/at-spi-bus-launcher
Debian+ 792 0.0  0.1 10272 3096 ttyl Ss+ 09:58 0:00 /usr/bin/dbus-daemon --session --nofork --print-address 11 -address
Debian+ 794 0.0  0.9 32256 40292 ttyl Ss+ 09:58 0:00 /usr/bin/kmod -W 1000 -L 1000 -notset accessx -core -auth /run/user/108/.mutter-Xwayland.0KWTZ -listenfd 4
Debian+ 796 0.0  0.2 23594 6808 ttyl Ss+ 09:58 0:00 /usr/libexec/at-spi2-registryd -use-gnome-session
colorbar 802 0.0  0.0 31792 6448 ? Ss+ 09:58 0:00 /usr/libexec/gdm-wayland-session
root 835 0.0  0.3 318496 7544 ? Ss+ 09:58 0:00 /usr/libexec/gpword
Debian+ 837 0.0  0.8 260218 16684 ttyl Ss+ 09:58 0:00 /usr/bin/gjs -m /usr/share/gnome-shell/org.gnome.Shell.Notifications
Debian+ 839 0.0  0.0 10272 3096 ? Ss 09:58 0:00 /usr/libexec/gdm-x11-kill-demon
Debian+ 843 0.0  0.6 412928 12156 ttyl Ss+ 09:58 0:00 /usr/libexec/gsd-wacom
Debian+ 858 0.0  0.6 414436 12984 ttyl Ss+ 09:58 0:00 /usr/libexec/gsd-color
Debian+ 853 0.0  0.6 485436 12484 ttyl Ss+ 09:58 0:00 /usr/libexec/gsd-keyboard
Debian+ 852 0.0  0.6 485436 12484 ttyl Ss+ 09:58 0:00 /usr/libexec/gsd-keyboard-notifications
Debian+ 873 0.0  0.2 53912 5296 ttyl Ss+ 09:58 0:00 /usr/libexec/gsd-rkill
Debian+ 876 0.0  0.2 459588 5972 ttyl Ss+ 09:58 0:00 /usr/libexec/gsd-smartcard
Debian+ 880 0.0  0.0 10272 3096 ? Ss 09:58 0:00 /usr/libexec/gsd-timer
Debian+ 892 0.0  0.7 747556 16812 ttyl Ss+ 09:58 0:00 /usr/libexec/gsd-netlink-keys
Debian+ 898 0.0  0.2 308720 5276 ttyl Ss+ 09:58 0:00 /usr/libexec/gsd-screensaver-proxy
Debian+ 904 0.0  0.0 10272 3096 ? Ss 09:58 0:00 /usr/libexec/gsd-screensaver
Debian+ 916 0.0  0.2 38316 4996 ttyl Ss+ 09:58 0:00 /usr/libexec/gsd-alloy-settings
Debian+ 919 0.0  0.3 458668 6792 ttyl Ss+ 09:58 0:00 /usr/libexec/gsd-housekeeping
Debian+ 928 0.0  0.7 525924 15284 ttyl Ss+ 09:58 0:00 /usr/libexec/gsd-power
Debian+ 934 0.0  0.0 10272 3096 ? Ss 09:58 0:00 /usr/libexec/gsd-power
Debian+ 1088 0.0  0.2 788788 54088 ttyl Ss+ 09:58 0:00 ibus-daemon --panel disable -r -xim
Debian+ 1099 0.0  0.4 461928 8272 ttyl Ss+ 09:58 0:00 ibus-daemon --panel disable -r -xim
Debian+ 1100 0.0  0.0 308928 13520 ttyl Ss+ 09:58 0:00 /usr/libexec/gdm-shutdown/gnome-shell.org.gnome.ScreenSaver
Debian+ 1105 0.0  0.3 389892 13520 ttyl Ss+ 09:58 0:00 /usr/libexec/gdm-doctor
Debian+ 1106 0.0  0.8 420952 17584 ttyl Ss+ 09:58 0:01 /usr/libexec/bus-extension-on-k3
Debian+ 1110 0.0  0.2 535792 36148 ttyl Ss+ 09:58 0:00 /usr/libexec/gdm-x11 -kill-demon
Debian+ 1111 0.0  0.3 253748 6240 ttyl Ss+ 09:58 0:00 /usr/libexec/gdm-x11-kill-demon
Debian+ 1113 0.0  0.3 235888 6240 ttyl Ss+ 09:58 0:00 /usr/libexec/gdm-engine-simple
root 1146 0.0  0.2 310232 5728 ? Ss 09:58 0:00 /usr/libexec/power-engine-daemon
Debian+ 1152 0.0  0.0 10272 3096 ? Ss 09:58 0:00 /usr/libexec/power-profile-service
root 1167 0.0  0.1 10184 2336 ? Ss+ 09:58 0:00 login - root
root 1188 0.0  0.3 21512 7496 ? Ss+ 09:58 0:00 /usr/lib/systemd/systemd -user
root 1190 0.0  0.0 10272 3096 ? Ss 09:58 0:00 /sbin/init
root 1195 0.0  0.0 8 0 ? Ss+ 09:58 0:00 /bin/sh
root 1206 0.0  0.0 5796 1588 ttyl Ss+ 09:58 0:00 /bin/bash
root 1208 0.0  0.0 2388 1296 ttyl Ss+ 09:58 0:00 /bin/sh /etc/X11/xserverrc .v12 _keepetty -auth /tmp/serverauth.X7oTp0Vb
root 1210 0.0  0.1 4584 2128 ttyl Ss+ 09:58 0:00 /init /etc/X11/xinit/xinitrc -- /etc/X11/xinit/xserverrc .v12 _keepetty -auth /tmp/serverauth.X7oTp0Vb

```

Рисунок 10 — `ps axu | more`

2.8. Отобрать из вывода команды `ps` строку, соответствующую процессу `dbus-daemon`, определить, где лежит её выполняемый файл и с какими параметрами он запущен

Для того чтобы отобрать необходимую строку используем команду `grep` совместно с командой `ps`:

```

root@pluto:~# ps axu|grep 'dbus-daemon'
message+ 549 0.0 0.2 8564 4744 ? Ss 10:01 0:00 /usr/bin/dbus-daemon --system --address=systemd: --nofork --nopidfile --systemd-activation --syslog-only
Debian+ 721 0.0 0.1 7680 2912 ? Ss 10:01 0:00 /usr/bin/dbus-daemon --session --address=systemd: --nofork --nopidfile --systemd-activation --syslog-only
debian+ 722 0.0 0.1 7680 3088 ? Ss 10:01 0:00 /usr/bin/dbus-daemon --nofork --print-address & session
root+ 792 0.0 0.1 7680 3088 ttysl S+ 10:01 0:00 /usr/bin/dbus-daemon --config-file=/usr/share/defaults/at-spi2/accessibility.conf --nofork --print-address 11 --address
=unix:/path/run/user/110/at-spi1/bus
root 1238 0.0 0.1 7400 3380 ? Ss 10:01 0:00 /usr/bin/dbus-daemon --session --address=systemd: --nofork --nopidfile --systemd-activation --syslog-only
root 3583 0.0 0.1 7400 3380 ? S 10:01 0:00 /usr/bin/dbus-daemon --config-file=/usr/share/defaults/at-spi2/accessibility.conf --nofork --print-address 10 --address
=unix:/path/run/user/0/at-spi1/bus
root 3586 0.0 0.0 6524 1824 pts/0 S+ 10:44 0:00 grep dbus-daemon
root@pluto:#

```

Рисунок 11 — `ps axu|grep 'dbus-daemon'`

2.9. Записать в файл с именем, содержащим текущее время, строку «-----» и список процессов

Напишем несколько команд, которые должны повторяться друг за другом в случае удачного/любого исхода предыдущей команды.

Вот

вся

комманда:

```
d=$(date); touch "$d"&&echo "dir:&&ls -la|grep \"$d\" ;echo
'-----'>>"$d"&&echo $(ps) >>"$d"&&echo "file:"&&cat "$d"
```

Разберем по порядку:

- `d=$(date)` берем текущее время в переменную `$d`
- `touch "$d"` создаем файл с названием в виде текущего времени
- `echo "dir:"` пишем о том, что следующим сведением в консоль поступит полная директория файла
- `ls -la|grep "$d"` ищем файл в текущей директории и выводим о нем информацию
- `echo '-----' >> "$d"` записываем в файл «-----»
- `echo $(ps) >> "$d"` Записываем в файл список текущих процессов
- `echo "file:"` Пишем что далее будет выведена информация записанная в файле
- `cat "$d"` читаем файл

```

root@pluto:~# ls -la
total 22980
drwxr-xr-x 22 root root 4096 Sep 13 11:56 .
drwxr-xr-x 18 root root 4096 Sep 13 11:56 ..
drwxr-xr-x 1 root root 208 Sep 9 21:22 .Xauthority
-rw-r--r-- 1 root root 6259 Sep 9 21:48 .bash_history
-rw-r--r-- 1 root root 4964 Sep 9 18:10 .bashrc
drwxr-xr-x 14 root root 4096 Sep 9 18:10 .cache
drwxr-xr-x 4 root root 4096 Sep 9 18:06 .corgi
drwxr-xr-x 16 root root 4096 Sep 9 10:20 .config
-rw-r--r-- 1 root root 2048 Sep 9 18:06 .deksrc0
-rw-r--r-- 1 root root 924 Sep 9 20:50 .deksrc1
drwxr-xr-x 2 root root 4096 Sep 9 20:14 .desktop
drwxr-xr-x 4 root root 4096 Sep 9 18:10 .local
drwxr-xr-x 1 root root 4096 Sep 9 21:38 .mozilla
-rw-r--r-- 1 root root 161 Aug 5 08:01 .profile
drwxr-xr-x 2 root root 4096 Sep 9 11:50 .ssh
drwxr-xr-x 2 root root 4096 Sep 9 10:19 .thumbnails
drwxr-xr-x 2 root root 4096 Sep 13 10:19 .w3m
-rw-r--r-- 1 root root 262 Sep 9 21:41 .wget-hsts
-rw-r--r-- 1 root root 41816 Sep 9 18:10 .xsession-errors
drwxr-xr-x 2 root root 4096 Sep 9 19:56 Desktop
drwxr-xr-x 2 root root 4096 Sep 9 19:56 Documents
drwxr-xr-x 3 root root 4096 Sep 9 19:56 Downloads
-rw-r--r-- 1 root root 2281446 Sep 9 18:23 gimp-codecs.zip
drwxr-xr-x 2 root root 4096 Sep 9 19:56 Music
drwxr-xr-x 2 root root 4096 Sep 9 19:56 Pictures
drwxr-xr-x 2 root root 4096 Sep 9 19:56 Public
drwxr-xr-x 2 root root 4096 Sep 9 19:56 Templates
drwxr-xr-x 2 root root 4096 Sep 9 19:56 Videos
drwxr-xr-x 2 root root 4096 Sep 9 19:56 .Xauthority
drwxr-xr-x 16 root root 4096 Sep 9 15:16 .dictionaries
drwxr-xr-x 7 user user 4096 Sep 9 14:40 mousepad-0.5.10
-rw-r--r-- 1 root root 959619 Jul 13 2022 mousepad-0.5.10.orig.tar.bz2
-rw-r--r-- 1 root root 2119 Sep 13 21:17 xorg.conf.new
root@pluto:~# touch "dir" && ls -la | grep "dir" >> "$d" && echo $ps >> "$d" && echo "file:" && cat "$d"
dir:
-rw-r--r-- 1 root root 0 Sep 13 11:05 Fri Sep 13 11:05:14 MSK 2024
files:
PID TIME CMD 1445 pts/0 00:00:01 bash 3264 pts/0 00:00:00 w3m 3272 pts/0 00:00:00 sh 3273 pts/0 00:00:00 w3m 3302 pts/0 00:00:00 sh 3303 pts/0 00:00:00
w3m 3315 pts/0 00:00:00 top 4263 pts/0 00:00:00 ps
root@pluto:~#

```

Рисунок 12 —

```

d=$(date);
touch "$d" &&
echo "dir:" &&
ls -la | grep "$d" ;
echo '-----' >> "$d" &&
echo $(ps) >> "$d" &&
echo "file:" &&
cat "$d"

```

2.10. Выполнить команду в фоновом режиме с отсрочкой запуска на 1 минуту. Продемонстрировать, что команда выполнилась именно через минуту.

```

(sleep 60; ls -la > output.txt) &
for i in {1..62}; do
    echo "Текущее время: $(date '+%Y-%m-%d %H:%M:%S')";
    [ -f output.txt ] && cat output.txt || 
    echo "Файл output.txt не найден";
    sleep 1;
done

```

Напишем еще одну составную команду:

- Отложим выполнение команды `ls -la > output.txt`, которая выводит информацию о текущей директории в файл `output.txt` на 1 минуту: `(sleep 60; ls -la > output.txt)`
- Организуем цикл, который в течении 62 секунд будет выводить текущее время и найден ли файл (пока мы не записали ничего файла не будет).

```

for i in {1..62}; do
    echo "Текущее время: $(date '+%Y-%m-%d %H:%M:%S')";
    [ -f output.txt ] && cat output.txt ||
    echo "Файл output.txt не найден";
    sleep 1;
done

```

```

[ Applications ] kitty
[ 1 ] * 2024-09-13 11:12:03
root@ipauto:~# [sleep 60; ls -l > output.txt] & for i in {1..62}; do echo "Текущее время: $(date '+%Y-%m-%d %H:%M:%S')"; [ -f output.txt ] && cat output.txt || echo "Файл output.txt не найден"; sleep 1; done
[16] 4998
Текущее время: 2024-09-13 11:12:03
Файл output.txt не найден
Текущее время: 2024-09-13 11:12:04
Файл output.txt не найден
Текущее время: 2024-09-13 11:12:05
Файл output.txt не найден
Текущее время: 2024-09-13 11:12:06
Файл output.txt не найден
Текущее время: 2024-09-13 11:12:07
Файл output.txt не найден
Текущее время: 2024-09-13 11:12:08
Файл output.txt не найден
Текущее время: 2024-09-13 11:12:09
Файл output.txt не найден
Текущее время: 2024-09-13 11:12:10
Файл output.txt не найден
Текущее время: 2024-09-13 11:12:11
Файл output.txt не найден
Текущее время: 2024-09-13 11:12:12
Файл output.txt не найден
Текущее время: 2024-09-13 11:12:13
Файл output.txt не найден
Текущее время: 2024-09-13 11:12:14
Файл output.txt не найден
Текущее время: 2024-09-13 11:12:15
Файл output.txt не найден
Текущее время: 2024-09-13 11:12:16
Файл output.txt не найден
Текущее время: 2024-09-13 11:12:17
Файл output.txt не найден
Текущее время: 2024-09-13 11:12:18
Файл output.txt не найден
Текущее время: 2024-09-13 11:12:19
Файл output.txt не найден
Текущее время: 2024-09-13 11:12:20
Файл output.txt не найден
Текущее время: 2024-09-13 11:12:21
Файл output.txt не найден
Текущее время: 2024-09-13 11:12:22
Файл output.txt не найден
Текущее время: 2024-09-13 11:12:23
Файл output.txt не найден
Текущее время: 2024-09-13 11:12:24
Файл output.txt не найден
Текущее время: 2024-09-13 11:12:25
Файл output.txt не найден
Текущее время: 2024-09-13 11:12:26
Файл output.txt не найден
Текущее время: 2024-09-13 11:12:27
Файл output.txt не найден

```

Рисунок 13 — Начало выполнения команды: минута еще не прошла

```

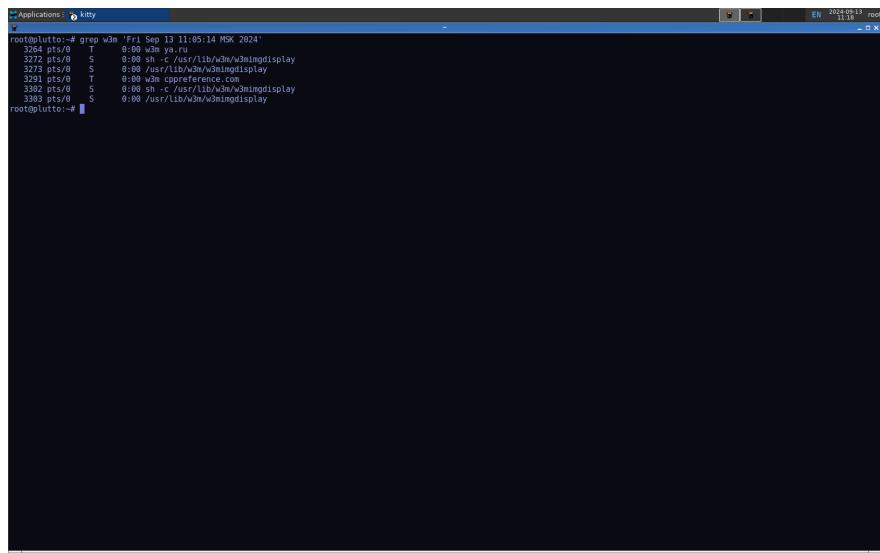
[ Applications ] kitty
[ 1 ] * 2024-09-13 11:12:39
root@ipauto:~# [sleep 60; ls -la > output.txt]
Файл output.txt не найден
Текущее время: 2024-09-13 11:12:40
Файл output.txt не найден
Текущее время: 2024-09-13 11:12:41
Файл output.txt не найден
Текущее время: 2024-09-13 11:12:42
Файл output.txt не найден
Текущее время: 2024-09-13 11:12:43
Файл output.txt не найден
Текущее время: 2024-09-13 11:12:44
Файл output.txt не найден
Текущее время: 2024-09-13 11:12:45
Файл output.txt не найден
[15] 22984
Текущее время: 2024-09-13 11:12:46
total 22984
drwxr-xr-x 2 root root 4096 Sep 13 11:12 .
drwxr-xr-x 10 root root 4096 Sep 9 11:50 ..
-rw-r--r-- 1 root root 200 Sep 9 21:22 .Xauthority
-rw-r--r-- 1 root root 625 Sep 9 18:00 .bash_history
-rw-r--r-- 1 root root 644 Sep 9 18:19 .bashrc
-rw-r--r-- 14 root root 4096 Sep 9 21:38 .cache
drwxr-xr-x 4 root root 4096 Sep 9 18:06 .cargo
drwxr-xr-x 10 root root 4096 Sep 9 18:06 .cargo_home
-rw-r--r-- 1 root root 4096 Sep 9 20:28 .gtrckr-z.0
-rw-r--r-- 3 root root 924 Sep 9 20:50 .indexrc
drwxr-xr-x 4 root root 4096 Sep 9 15:03 .local
drwxr-xr-x 4 root root 4096 Sep 9 21:38 .mozilla
drwxr-xr-x 3 root root 4096 Sep 9 10:58 .profile
drwxr-xr-x 3 root root 4096 Sep 9 11:50 .themes
drwxr-xr-x 2 root root 4096 Sep 13 10:19 .w2m
-rw-r--r-- 1 root root 300 Sep 9 10:00 .xsession-errors
-rw-r--r-- 1 root root 420869 Sep 13 11:09 .xsession-errors
drwxr-xr-x 2 root root 4096 Sep 9 10:56 Desktop
drwxr-xr-x 2 root root 4096 Sep 9 10:56 Documents
drwxr-xr-x 2 root root 4096 Sep 9 19:56 Downloads
-rw-r--r-- 1 root root 2201448 Jun 4 2023 FiraCode.zip
drwxr-xr-x 1 root root 259 Sep 13 10:05 Fri Sep 13 11:05:14 MSK 2024
drwxr-xr-x 2 root root 4096 Sep 9 10:56 Pictures
drwxr-xr-x 2 root root 4096 Sep 9 19:56 Pictures
drwxr-xr-x 2 root root 4096 Sep 9 19:56 Public
drwxr-xr-x 2 root root 4096 Sep 9 19:56 Shared
drwxr-xr-x 2 root root 4096 Sep 9 19:56 Videos
drwxr-xr-x 7 root root 4096 Sep 9 20:41 aesthetic-wallpapers
drwxr-xr-x 10 root root 4096 Sep 9 14:40 dottiles
drwxr-xr-x 1 root root 4096 Sep 9 14:40 mousepad-0.5.10.tar.bz2
-rw-r--r-- 1 root root 959696 Jul 13 2022 mousepad-0.5.10.orig.tar.bz2
-rw-r--r-- 1 root root 0 Sep 13 11:12 output.txt
-rw-r--r-- 1 root root 2119 Sep 9 21:17 xorg.conf.new

```

Рисунок 14 — Комманда выполнилась, файл найден и выведен в терминал

2.11. Отобрать из одного из сформированных файлов строки, относящиеся к одному из процессов.

Чтобы отобрать строки, относящиеся к одному процессу напишем комманду `grep` и передадим в качестве аргумента название процесса и название файла:



```
root@pluto:~# grep w3m 'Fri Sep 13 11:05:14 MSK 2024'
3264 pts/0    T    0:00 w3m ya.ru
3272 pts/0    S    0:00 sh -c /usr/lib/w3m/v3mimgdisplay
3273 pts/0    S    0:00 sh -c /usr/lib/w3m/v3mimgdisplay
3293 pts/0    S    0:00 w3m cpreference.com
3302 pts/0    S    0:00 sh -c /usr/lib/w3m/v3mimgdisplay
3303 pts/0    S    0:00 /usr/lib/w3m/v3mimgdisplay
root@pluto:~#
```

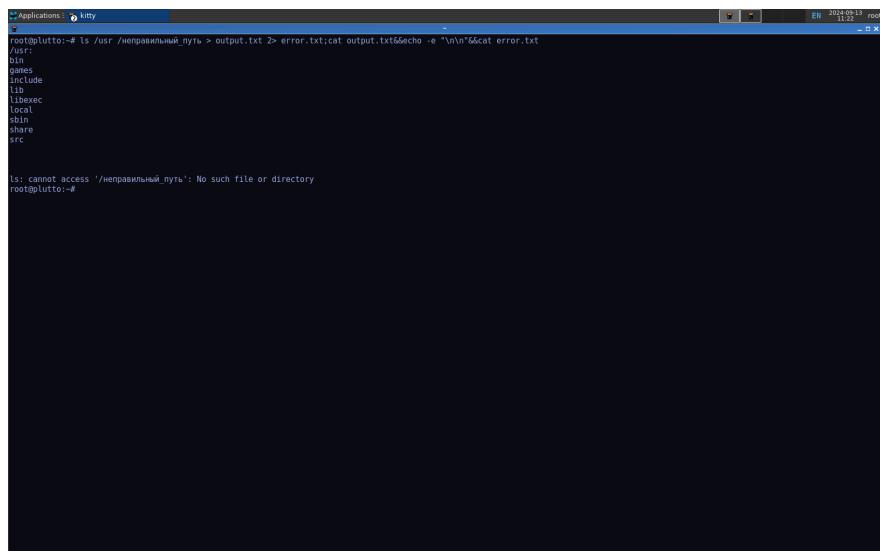
Рисунок 15 — `grep w3m 'Fri Sep 13 11:05:14 MSK 2024'`

2.12. Вывести результаты работы произвольной команды в один файл, а сообщения об ошибках в другой. Продемонстрировать правильность работы.

Для выполнения данной операции воспользуемся следующей командой: `ls /usr /неправильный_путь > output.txt 2> error.txt`

Эта команда выведет все, что есть в каталоге `/usr` в один файл и ошибку об отсутствии каталога `/неправильный_путь` в другой.

Дополнительно к команде припишем чтение этих 2 файлов через отступ.



```
root@pluto:~# ls /usr /неправильный_путь > output.txt 2> error.txt;cat output.txt&&echo -e "\n\n"&&cat error.txt
bin
games
include
lib
libexec
local
sbin
share
src

ls: cannot access '/неправильный_путь': No such file or directory
root@pluto:~#
```

Рисунок 16 —

```
ls /usr /неправильный_путь > output.txt 2> error.txt;
cat output.txt &&
echo -e "\n\n" &&
cat error.txt
```

2.13. Выполнить произвольную команду с ограничением использования процессорного времени 300 секунд и выводом результатов и сообщений об ошибках в один файл.

Команда `find` может занять достаточно много времени, поэтому ограничим ее выполнение до 300с. Вывод направим в файл `output.txt`, после чего прочитаем последние 10 строк из него: `timeout 300s find / > output.txt 2>&1; tail output.txt`

2.14. Настроить cron на выполнение команды ежедневно в заданное время. Продемонстрировать правильность работы.

Для начала откроем конфиг-файл `cron`:

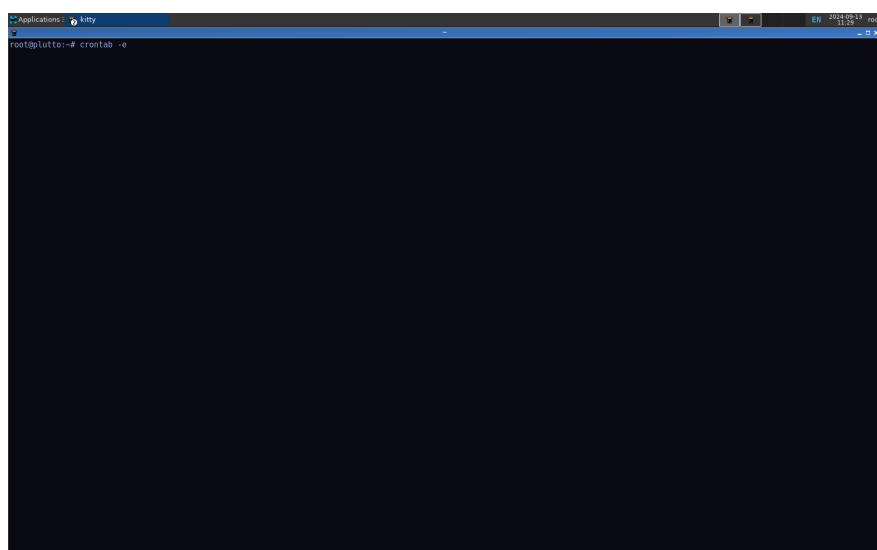


Рисунок 17 — `crontab -e`

Пропишем сюда следующую строку:

```
42 11 * * * /usr/bin/env echo "Hello World" » /output.txt
```

- `42 11` Означает, что команда выполнится в 11:42.
- `* * *` Означают, что команда будет выполняться каждый день
- `/usr/bin/env` Путь к среде
- `echo "Hello World" » /output.txt` Комманда

```
# Edit this file to introduce tasks to be run by cron.
# Each task to run has to be defined through a single line
# indicating with different fields when the task will be run
# and what command to run for the task
#
# To define the time you can provide concrete values for
# minute (min), hour (hour), month (month), year
# and day of week (dow) or use "*" in these fields (for "any").
#
# Notice that tasks will be started based on the cron's system
# daemon's notion of time and timezones.
#
# Output of the cron jobs (including errors) is sent through
# email to the user the crontab file belongs to (unless redirected).
#
# For example, you can run a backup of all your user accounts
# every day at 1:55 am:
# 0 5 * * * tar -cf /var/backups/home.tgz /home/
#
# For more information see the manual pages of crontab(5) and cron(8)
#
# m h dom mon dow   command
38 11 * * * /usr/bin/env echo "Hello" >> /output.txt
```

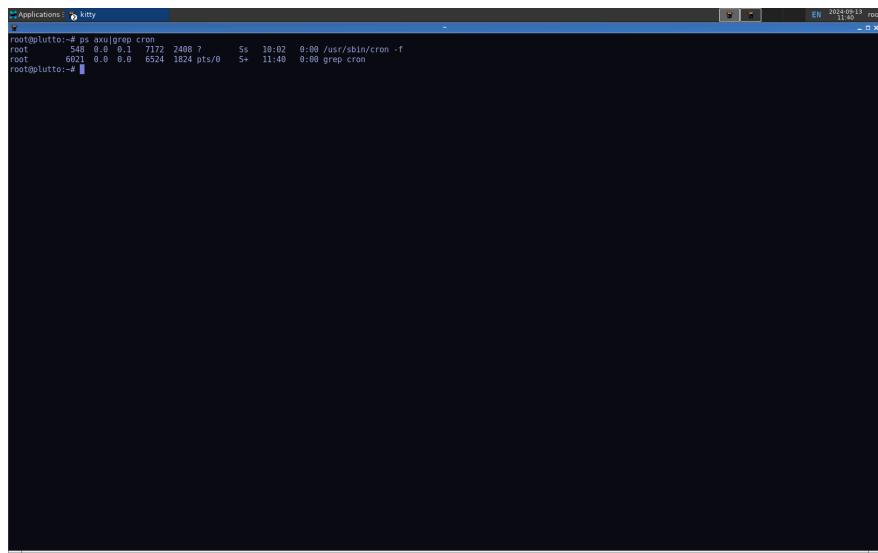
Рисунок 18 — `crontab -e`

Проверим все ли правильно записалось:

```
root@pluto:~# crontab -e
crontab: installing new crontab
root@pluto:~# crontab -l
# Edit this file to introduce tasks to be run by cron.
#
# Each task to run has to be defined through a single line
# indicating with different fields when the task will be run
# and what command to run for the task
#
# To define the time you can provide concrete values for
# minute (min), hour (hour), month (month), year
# and day of week (dow) or use "*" in these fields (for "any").
#
# Notice that tasks will be started based on the cron's system
# daemon's notion of time and timezones.
#
# Output of the cron jobs (including errors) is sent through
# email to the user the crontab file belongs to (unless redirected).
#
# For example, you can run a backup of all your user accounts
# every day at 1:55 am:
# 0 5 * * * tar -cf /var/backups/home.tgz /home/
#
# For more information see the manual pages of crontab(5) and cron(8)
#
# m h dom mon dow   command
42 11 * * * /usr/bin/env echo "Hello" >> /output.txt
```

Рисунок 19 — `crontab -l`

Проверим запущен ли демон `cron` :

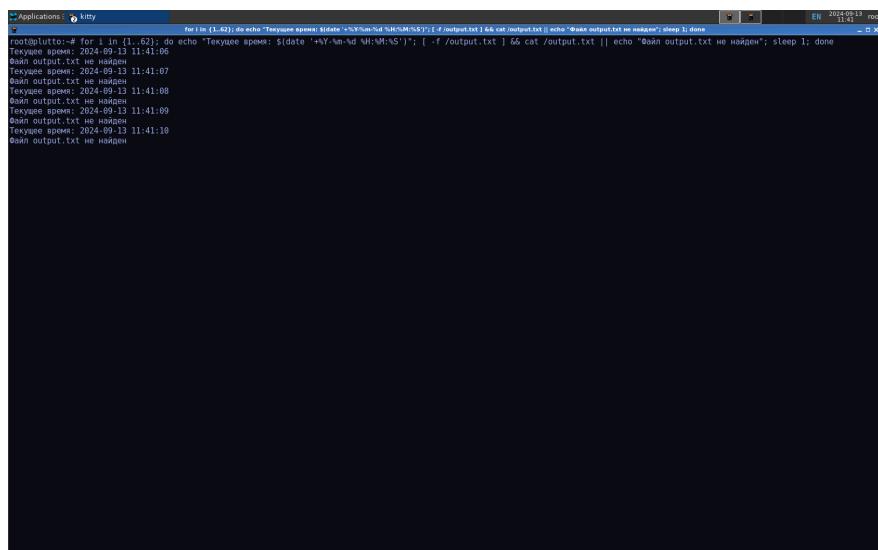


```
root@pluto:~# ps aux|grep cron
root      548  0.0  0.1   7172  2408 ?        Ss   10:02  0:00 /usr/sbin/cron -f
root     6021  0.0  0.0  6524 1824 pts/0    S+   11:40  0:00 grep cron
```

Рисунок 20 — `ps aux|grep cron`

За минуту до назначенного времени запустим команду, которая каждую секунду в течении 62 секунд проверяет наличие файла `output.txt` и его содержимое:

```
for i in {1..62}; do
echo "Текущее время: $(date '+%Y-%m-%d %H:%M:%S')";
[ -f output.txt ] &&
cat output.txt ||
echo "Файл output.txt не найден";
sleep 1;
done
```



```
root@pluto:~# for i in {1..62}; do echo "Текущее время: $(date '+%Y-%m-%d %H:%M:%S')"; [ -f output.txt ] && cat output.txt || echo "Файл output.txt не найден"; sleep 1; done
Текущее время: 2024-09-13 11:41:06
Файл output.txt не найден
Текущее время: 2024-09-13 11:41:07
Файл output.txt не найден
Текущее время: 2024-09-13 11:41:08
Файл output.txt не найден
Текущее время: 2024-09-13 11:41:09
Файл output.txt не найден
Текущее время: 2024-09-13 11:41:10
Файл output.txt не найден
```

Рисунок 21 — Комманда

```
Applications / kity
EN 2024-09-13 root 31.42 - x
Файл output.txt не найден
Текущее время: 2024-09-13 11:41:44
Файл output.txt не найден
Текущее время: 2024-09-13 11:41:45
Файл output.txt не найден
Текущее время: 2024-09-13 11:41:46
Файл output.txt не найден
Текущее время: 2024-09-13 11:41:47
Файл output.txt не найден
Текущее время: 2024-09-13 11:41:48
Файл output.txt не найден
Текущее время: 2024-09-13 11:41:49
Файл output.txt не найден
Текущее время: 2024-09-13 11:41:50
Файл output.txt не найден
Текущее время: 2024-09-13 11:41:51
Файл output.txt не найден
Текущее время: 2024-09-13 11:41:52
Файл output.txt не найден
Текущее время: 2024-09-13 11:41:53
Файл output.txt не найден
Текущее время: 2024-09-13 11:41:54
Файл output.txt не найден
Текущее время: 2024-09-13 11:41:55
Файл output.txt не найден
Текущее время: 2024-09-13 11:41:56
Файл output.txt не найден
Текущее время: 2024-09-13 11:41:57
Файл output.txt не найден
Текущее время: 2024-09-13 11:41:58
Файл output.txt не найден
Текущее время: 2024-09-13 11:41:59
Файл output.txt не найден
Текущее время: 2024-09-13 11:42:00
Hello
Hello
Текущее время: 2024-09-13 11:42:01
Hello
Hello
Текущее время: 2024-09-13 11:42:02
Hello
Hello
Текущее время: 2024-09-13 11:42:03
Hello
Текущее время: 2024-09-13 11:42:04
Hello
Hello
Текущее время: 2024-09-13 11:42:05
Hello
Hello
Текущее время: 2024-09-13 11:42:06
Hello
Hello
Текущее время: 2024-09-13 11:42:07
Hello
Hello
Текущее время: 2024-09-13 11:42:08
Hello
Hello
root@kali:~#
```

Рисунок 22 — Комманда

В 11:42 файл появился и в него записалось Hello.