

Лабораторная работа 5

Тема: Загрузка ОС и процессы

Цель: ознакомиться с процессами загрузки операционной системы Linux и основными аспектами управления процессами.

Задание:

1. Потоки ввода/вывода.

- создать файл, используя команду `echo`.

- используя команду `ls`, прочитать содержимое каталога `etc` в файл, ошибки перенаправить в отдельный файл.

2. Конвейер (pipeline). Используйте команду **awk** на вывод длинного списка каталога, чтобы отобразить только права доступа к файлам. Затем отправьте в конвейере этот вывод на **sort** и **uniq**, чтобы отсортировать и отфильтровать все повторяющиеся строки.

3. Управление процессами. Создайте пустой файл, например **hello.sh**, откройте его в редакторе и напишите простой bash-скрипт:

```
GNU nano 6.2
#!/bin/bash
echo "Hello, World!"
while true; do
    sleep 10
done
```

Сохраните изменения, сделайте файл исполняемым и запустите его. Для запуска в фоновом режиме используется амперсанд **&** после команды, например: **./hello.sh &**

Убедитесь, что процесс отображается при выполнении команды **ps**, затем завершите его командой **kill**. Убедитесь, что процесс завершен.

Сделайте тоже самое, только на Python. Проверьте, что Python установлен:

```
$ python3 -V
```

Создайте файл, например, **hello.py**, напишите простой скрипт, например:

```
GNU nano 6.2
#!/usr/bin/env python3
import time
print("Hello!")
def main():
    while True:
        time.sleep(1)

if __name__ == "__main__":
    main()
```

Сделайте файл исполняемым, запустите его в фоновом режиме, убедитесь что процесс выполняется и “убейте” его вручную.

4. Управление процессами. Установите веб-сервер nginx и запустите его:

```
$ sudo apt install nginx
```

```
$ sudo nginx
```

Определите его PID (используйте **ps** и **grep**) и завершите командой **kill**. Сначала попробуйте корректное завершение. Убедитесь, что процесс завершен.

5. Управление процессами. Запустите nginx:

```
$ sudo service nginx start
```


Узнайте IP-адрес сервера:

```
$ ip a
```

Откройте браузер, в строке поиска введите полученный адрес, должна открыться стартовая страница nginx.

Отредактируйте стартовую страницу (/var/www/html/index.html)

Перезапустите nginx и обновите страницу в браузере:

 192.168.1.162

Welcome to Laboratory work N5!

Welcome to Laboratory work N 5

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

Thank you for using nginx.

6. Изменение конфигурационных файлов служб. Откройте конфигурационный файл nginx (/etc/nginx/sites-enabled/@default) и измените следующие строки, поменяв тем самым порт по умолчанию (80) на кастомный (в примере 8080):

```
# Default server configuration
#
server {
    listen 8080 default_server;
    listen [::]:8080 default_server;
```

Перезапустите nginx:

```
$ sudo service nginx restart
```

Обновите страницу в браузере. Потом попробуйте открыть страницу добавив к адресу новый порт 8080:



192.168.1.162:8080

Welcome to Laboratory work N5!

Welcome to Laboratory work N 5

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

Thank you for using nginx.

7. Автозагрузка скриптов.

Измените скрипт `hello.cs`. Эта команда `echo` отправляет сообщение "Hello World!" в системный консольный вывод (`/dev/console`), который должен быть виден в консоли после автозагрузки.

```
GNU nano 6.2
#!/bin/bash
echo "Hello, World!" >> /dev/console
while true; do
    sleep 10
done
```

Создайте systemd-юнита, например `hello.service` в директории `/etc/systemd/system/`:

```
GNU nano 6.2 /etc/systemd/system/hello.service
[Unit]
Description=Hello Script
After=multi-user.target

[Service]
Type=idle
ExecStart=/home/svv/hello.sh

[Install]
WantedBy=default.target
```

Убедитесь, что у файла юнита есть правильные права доступа:

`sudo chmod 644 /etc/systemd/system/hello.service`

Обновите список юнитов systemd:

`sudo systemctl daemon-reload`

Включите автозапуск вашего сервиса:

`sudo systemctl enable hello.service`

Убедитесь, что ваш скрипт автоматически запускается при загрузке системы.

Управление службой:

- Запустить службу: **`sudo systemctl start hello`**
- Остановить службу: **`sudo systemctl stop hello`**
- Перезапустить службу: **`sudo systemctl restart hello`**