

Лабораторная работа №10

НКАБд-06-23

Улитина Мария Максимовна

Содержание

1 Цель работы.....	1
2 Задание	1
3 Теоретическое введение	2
3.1 Права доступа к файлам	2
3.2 Работа с файлами средствами NASM	2
3.2.1 Открытие и создание файла	3
3.2.2 Запись в файл	3
3.2.3 Чтение файла.....	3
3.2.4 Заккрытие файла.....	3
3.2.5 Изменение содержимого файла	3
3.2.6 Удаление файла	3
4 Выполнение лабораторной работы	3
4.1 Задание для самостоятельной работы	6
5 Выводы	6
Список литературы.....	7

1 Цель работы

Приобретение навыков написания программ для работы с файлами.

2 Задание

1. Создание каталога для программ лабораторной работы №10 и создание необходимых файлов.

2. Введение в файл текста программы и её исполнение.
3. Изменение прав доступа к файлам.
4. Написание программы, создающей файл.

3 Теоретическое введение

3.1 Права доступа к файлам

ОС GNU/Linux является многопользовательской операционной системой. И для обеспечения защиты данных одного пользователя от действий других пользователей существуют специальные механизмы разграничения доступа к файлам. Кроме ограничения доступа, данный механизм позволяет разрешить другим пользователям доступ данным для совместной работы. Права доступа определяют набор действий (чтение, запись, выполнение), разрешённых для выполнения пользователям системы над файлами. Для каждого файла пользователь может входить в одну из трех групп: владелец, член группы владельца, все остальные. Для каждой из этих групп может быть установлен свой набор прав доступа. Владелцем файла является его создатель. Набор прав доступа задается тройками битов и состоит из прав на чтение, запись и исполнение файла. В символьном представлении он имеет вид строк `gwx`, где вместо любого символа может стоять дефис. Буква означает наличие права (установлен в единицу второй бит триады `g` — чтение, первый бит `w` — запись, нулевой бит `x` — исполнение), а дефис означает отсутствие права (нулевое значение соответствующего бита). Также права доступа могут быть представлены как восьмеричное число. Так, права доступа `gw-` (чтение и запись, без исполнения) понимаются как три двоичные цифры `110` или как восьмеричная цифра `6`.

3.2 Работа с файлами средствами NASM

В операционной системе Linux существуют различные методы управления файлами, например, такие как создание и открытие файла, только для чтения или для чтения и записи, добавления в существующий файл, закрытия и удаления файла, предоставление прав доступа.

Обработка файлов в операционной системе Linux осуществляется за счет использования определенных системных вызовов. Для корректной работы и доступа к файлу при его открытии или создании, файлу присваивается уникальный номер (16-битное целое число) - дескриптор файла.

3.2.1 Открытие и создание файла

Для создания и открытия файла служит системный вызов `sys_creat`, который использует следующие аргументы: права доступа к файлу в регистре `ECX`, имя файла в `EBX` и номер системного вызова (8) в `EAX`.

3.2.2 Запись в файл

Для записи в файл служит системный вызов `sys_write`, который использует следующие аргументы: количество байтов для записи в регистре `EDX`, строку содержимого для записи в `ECX`, файловый дескриптор в `EBX` и номер системного вызова (4) в `EAX`.

3.2.3 Чтение файла

Для чтения данных из файла служит системный вызов `sys_read`, который использует следующие аргументы: количество байтов для чтения в регистре `EDX`, адрес в памяти для записи прочитанных данных в `ECX`, файловый дескриптор в `EBX` и номер системного вызова (3) в `EAX`.

3.2.4 Заккрытие файла

Для правильного закрытия файла служит системный вызов `sys_close`, который использует один аргумент – дескриптор файла в регистре `EBX`. После вызова ядра происходит удаление дескриптора файла, а в случае ошибки, системный вызов возвращает код ошибки в регистр `EAX`.

3.2.5 Изменение содержимого файла

Для изменения содержимого файла служит системный вызов `sys_lseek`, который использует следующие аргументы: исходная позиция для смещения в `EDX`, значение смещения в байтах в `ECX`, файловый дескриптор в `EBX` и номер системного вызова `sys_lseek` в `EAX`.

3.2.6 Удаление файла

Удаление файла осуществляется системным вызовом `sys_unlink`, который использует один аргумент – имя файла в регистре `EBX`.

4 Выполнение лабораторной работы

Создадим каталог для программ лабораторной работы и создадим в нем необходимые файлы (рис. 1).

```
mmulitina@ubuntu:~$ mkdir ~/work/arch-pc/lab10
mmulitina@ubuntu:~$ cd ~/work/arch-pc/lab10
mmulitina@ubuntu:~/work/arch-pc/lab10$ touch lab10-1.asm readme-1.txt readme-2.txt
```

Figure 1: Создание каталога

Введём в файл `lab10-1.asm` текст программы из листинга (рис. 2).

```

#include 'in_out.asm'
SECTION .data
filename db 'readme.txt', 0h ; Имя файла
msg db 'Введите строку для записи в файл: ', 0h ; Сообщение
SECTION .bss
contents resb 255 ; переменная для вводимой строки
SECTION .text
global _start
_start:
; --- Печать сообщения `msg`
mov eax,msg
call sprint
; ---- Запись введенной с клавиатуры строки в `contents`
mov ecx, contents
mov edx, 255
call sread
; --- Открытие существующего файла (`sys_open`)
mov ecx, 2 ; открываем для записи (2)
mov ebx, filename
mov eax, 5

```

Figure 2: Ввод программы

Создадим исполняемый файл и проверим его работу (рис. 3).

```

mmulitina@ubuntu:~/work/arch-pc/lab10$ nasm -f elf lab10-1.asm
mmulitina@ubuntu:~/work/arch-pc/lab10$ ld -m elf_i386 -o lab10-1 lab10-1.o
mmulitina@ubuntu:~/work/arch-pc/lab10$ ./lab10-1
Введите строку для записи в файл: Hello

```

Figure 3: Проверка файла

(рис. 4).

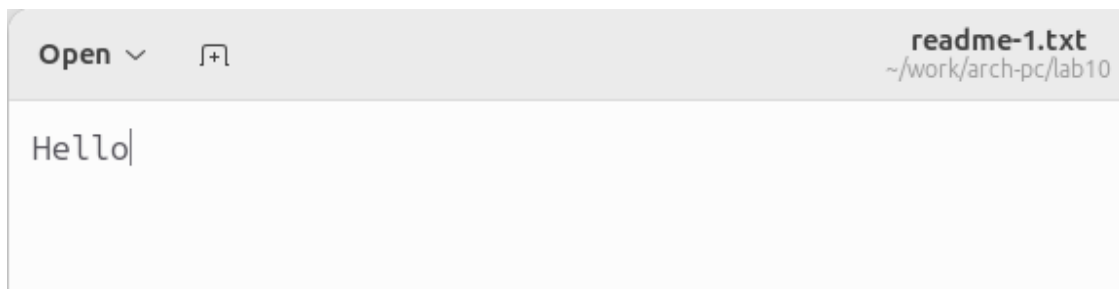


Figure 4: Проверка файла

С помощью команды `chmod` изменим права доступа к исполняемому файлу `lab10-1`, запретив его выполнение (рис. 5).

```
mmulitina@ubuntu:~/work/arch-pc/lab10$ chmod u-x lab10-1
mmulitina@ubuntu:~/work/arch-pc/lab10$ ./lab10-1
bash: ./lab10-1: Permission denied
```

Figure 5: Отказ в доступе к файлу

Доступ к файлу запрещён.

Изменим права доступа к файлу lab10-1.asm, добавив права на исполнение (рис. 6).

```
mmulitina@ubuntu:~/work/arch-pc/lab10$ chmod u+x lab10-1.asm
mmulitina@ubuntu:~/work/arch-pc/lab10$ ./lab10-1.asm
./lab10-1.asm: line 1: fg: no job control
./lab10-1.asm: line 2: SECTION: command not found
./lab10-1.asm: line 3: filename: command not found
./lab10-1.asm: line 3: Имя: command not found
./lab10-1.asm: line 4: msg: command not found
./lab10-1.asm: line 4: Сообщение: command not found
./lab10-1.asm: line 5: SECTION: command not found
./lab10-1.asm: line 6: contents: command not found
./lab10-1.asm: line 6: переменная: command not found
./lab10-1.asm: line 7: SECTION: command not found
./lab10-1.asm: line 8: global: command not found
./lab10-1.asm: line 9: _start:: command not found
./lab10-1.asm: line 10: syntax error near unexpected token `;'
./lab10-1.asm: line 10: `; --- Печать сообщения `msg`'
```

Figure 6: Изменение прав доступа к файлу

Файл нельзя выполнить, т.к. он не содержит указаний для терминала, а просто является текстом программы.

Предоставим права доступа к файлам в соответствии с вариантом 3 (рис. 7).

```
mmulitina@ubuntu:~/work/arch-pc/lab10$ chmod u=rx readme-1.txt
mmulitina@ubuntu:~/work/arch-pc/lab10$ chmod g=wx readme-1.txt
mmulitina@ubuntu:~/work/arch-pc/lab10$ chmod o=rw readme-1.txt
mmulitina@ubuntu:~/work/arch-pc/lab10$ ls -l readme-1.txt
-r-x-wxrw- 1 mmulitina mmulitina 6 Dec  8 05:11 readme-1.txt
```

Figure 7: Изменение прав доступа к файлу

(рис. 8).

```
mmulitina@ubuntu:~/work/arch-pc/lab10$ chmod 353 readme-2.txt
mmulitina@ubuntu:~/work/arch-pc/lab10$ ls -l readme-2.txt
--wxr-x-wx 1 mmulitina mmulitina 0 Dec  8 04:54 readme-2.txt
```

Figure 8: Изменение прав доступа к файлу

4.1 Задание для самостоятельной работы

Напишем программу, работающую по представленному алгоритму (рис. 9).

```
mmulitina@ubuntu:~/work/arch-pc/lab10$ nasm -f elf name.asm
mmulitina@ubuntu:~/work/arch-pc/lab10$ ld -m elf_i386 -o name name.o
mmulitina@ubuntu:~/work/arch-pc/lab10$ ./name
Как Вас зовут?Maria
```

Figure 9: Выполнение программы

Проверим наличие файла (рис. 10).

```
mmulitina@ubuntu:~/work/arch-pc/lab10$ ls
in_out.asm  lab10-1.asm  name        name.o      readme-1.txt
lab10-1     lab10-1.o   name.asm    name.txt    readme-2.txt
```

Figure 10: Проверка наличия файла

И его содержание (рис. 11).

```
mmulitina@ubuntu:~/work/arch-pc/lab10$ cat name.txt
.Меня зовут Maria
```

Figure 11: Проверка содержания файла

5 Выводы

В процессе выполнения лабораторной работы я приобрела навыки написания программ для работы с файлами.

Список литературы

Лабораторная работа №10.