

Отчёт по лабораторной работе №10

Работа с файлами средствами Nasm

Корчагин Алексей Павлович

Содержание

1	Цель работы	5
2	Теоретическое введение	6
3	Выполнение лабораторной работы	7
4	Выполнение самостоятельной работы	12
5	Выводы	17
	Список литературы	18

Список иллюстраций

3.1	Создание файлов	7
3.2	Редактирование файла	8
3.3	Создание исполняемого файла	8
3.4	Работа программы	9
3.5	Изменение прав	9
3.6	Присвоение прав	10
3.7	Присвоение прав	10
3.8	Присвоение прав	11
4.1	Редактирование файла	13
4.2	Исполняемый файл	13
4.3	Проверка содержимого	14

Список таблиц

1 Цель работы

Преобрести навыки написания программ для работы с файлами NASM

2 Теоретическое введение

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.

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. Всего возможно 8 комбинаций, приведенных в таблице 10.1. Буква означает наличие права (установлен в единицу второй бит триады `r` — чтение, первый бит `w` — запись, нулевой бит `x` — исполнение), а дефис означает отсутствие права (нулевое значение соответствующего бита).

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 Выполнение лабораторной работы

С помощью `mkdir` создаю директорию `lab10`, перехожу в нее и создаю файл для работы(рис. 3.1).

```
apkorchagin@dk2n26 ~ $ mkdir ~/work/arch-pc/lab10 && cd ~/work/arch-pc/lab10 &&  
touch lab10-1.asm readme-1.txt readme-2.txt  
apkorchagin@dk2n26 ~/work/arch-pc/lab10 $ █
```

Рис. 3.1: Создание файлов

Открываю созданный файл `lab10-1.asm`, вставляю в него программу из Листинга.(рис. 3.2).

```

1 %include 'in_out.asm'
2 SECTION .data
3 filename db 'readme.txt', 0h ; Имя файла
4 msg db 'Введите строку для записи в файл: ', 0h ; Сообщение
5 SECTION .bss
6 contents resb 255 ; переменная для вводимой строки
7 SECTION .text
8 global _start
9 _start:
10 ; --- Печать сообщения 'msg'
11 mov eax,msg
12 call sprint
13 ; ---- Запись введенной с клавиатуры строки в 'contents'
14 mov ecx, contents
15 mov edx, 255
16 call sread
17 ; --- Открытие существующего файла ('sys_open')
18 mov ecx, 2 ; открываем для записи (2)
19 mov ebx, filename
20 mov eax, 5
21 int 80h
22 ; --- Запись дескриптора файла в 'esi'
23 mov esi, eax
24 ; --- Расчет длины введенной строки
25 mov eax, contents ; в 'eax' запишется количество
26 call slen ; введенных байтов
27 ; --- Записываем в файл 'contents' ('sys_write')
28 mov edx, eax
29 mov ecx, contents
30 mov ebx, esi
31 mov eax, 4
32 int 80h
33 ; --- Закрываем файл ('sys_close')
34 mov ebx, esi
35 mov eax, 6
36 int 80h
37 call quit
38

```

Рис. 3.2: Редактирование файла

Создаю исполняемый файл программы, а также сам файл, куда будет записываться строка. (рис. 3.3).

```

аркorchаgin@dk2n26 ~/work/arch-pc/lab10 $ nasm -f elf lab10-1.asm && ld -m elf_i
386 -o lab10-1 lab10-1.o
аркorchаgin@dk2n26 ~/work/arch-pc/lab10 $ touch readme.txt && ./lab10-1
Введите строку для записи в файл: Алексей Корчагин
аркorchаgin@dk2n26 ~/work/arch-pc/lab10 $

```

Рис. 3.3: Создание исполняемого файла

Проверяю работу программы.(рис. 3.4).

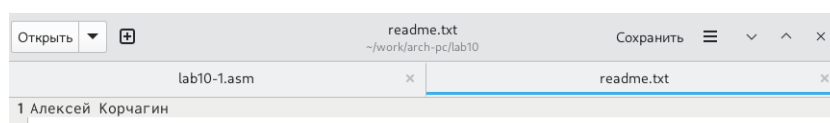


Рис. 3.4: Работа программы

С помощью команды `chmod` изменяю права доступа к исполняемому файлу `lab10-1`, запретив его выполнение.(рис. 3.5).

```
apkorchagin@dk2n26 ~/.work/arch-pc/lab10 $ chmod 640 lab10-1
apkorchagin@dk2n26 ~/.work/arch-pc/lab10 $ ./lab10-1
bash: ./lab10-1: Отказано в доступе
apkorchagin@dk2n26 ~/.work/arch-pc/lab10 $
```

Рис. 3.5: Изменение прав

В символьном представлении существует возможность явно указывать в какой группе какие права необходимо добавить или отменить. Так '6' дает право владельцу лишь на чтение и запись, но не на исполнения. '4' дает право группе лишь на чтения. '0' оменяет все права для всех прочих пользователей.

С помощью команды `chmod` изменяю права доступа к файлу `lab10-1.asm` с исходным текстом программы, добавив права на исполнение. (рис. 3.6).

```

apkorchagin@dk2n26 ~/work/arch-pc/lab10 $ chmod u+x lab10-1.asm
apkorchagin@dk2n26 ~/work/arch-pc/lab10 $ ./lab10-1
bash: ./lab10-1: Отказано в доступе
apkorchagin@dk2n26 ~/work/arch-pc/lab10 $ ./lab10-1.asm
./lab10-1.asm: строка 1: fg: нет управления заданиями
./lab10-1.asm: строка 2: SECTION: команда не найдена
./lab10-1.asm: строка 3: filename: команда не найдена
./lab10-1.asm: строка 3: Имя: команда не найдена
./lab10-1.asm: строка 4: msg: команда не найдена
./lab10-1.asm: строка 4: Сообщение: команда не найдена
./lab10-1.asm: строка 5: SECTION: команда не найдена
./lab10-1.asm: строка 6: contents: команда не найдена
./lab10-1.asm: строка 6: переменная: команда не найдена
./lab10-1.asm: строка 7: SECTION: команда не найдена
./lab10-1.asm: строка 8: global: команда не найдена
./lab10-1.asm: строка 9: _start:: команда не найдена
./lab10-1.asm: строка 10: синтаксическая ошибка рядом с неожиданным маркером «;»
./lab10-1.asm: строка 10: `; --- Печать сообщения `msg`
apkorchagin@dk2n26 ~/work/arch-pc/lab10 $

```

Рис. 3.6: Присвоение прав

С помощью аргументов “u” и “+” добавляем право владельцу на исполнение файла, однако в файле находится исходный код программы, который еще не ассемблирован, и потому его выполнение невозможно. Происходит исполнение файла, но программа работать не будет.

В соответствии со своим 2-ым вариантом, предоставляю права доступа к файлу readme-1.txt представленные в символьном виде: rwx rwx -x. Вначале проверяю права, которые имеют пользователи при работе с файлом по умолчанию, затем даю ей другие права в соответствии с вариантом. (рис. 3.7).

```

apkorchagin@dk2n26 ~/work/arch-pc/lab10 $ ls -l readme-1.txt
-rw-r--r-- 1 apkorchagin studsci 0 дек 16 09:59 readme-1.txt
apkorchagin@dk2n26 ~/work/arch-pc/lab10 $ chmod 771 readme-1.txt
apkorchagin@dk2n26 ~/work/arch-pc/lab10 $ ls -l readme-1.txt
-rwxrwx--x 1 apkorchagin studsci 0 дек 16 09:59 readme-1.txt

```

Рис. 3.7: Присвоение прав

Предоставляю права доступа к файлу readme-2.txt представленные в символьном – в двоичном виде: 110 111 101, то есть 6 7 5 (рис. 3.8).

```
apkorchagin@dk2n26 ~/work/arch-pc/lab10 $ chmod 675 readme-2.txt
apkorchagin@dk2n26 ~/work/arch-pc/lab10 $ ls -l readme-2.txt
-rw-rwxr-x 1 apkorchagin studsci 0 дек 16 09:59 readme-2.txt
apkorchagin@dk2n26 ~/work/arch-pc/lab10 $
```

Рис. 3.8: Присвоение прав

4 Выполнение самостоятельной работы

Создал новый файл. Написал в его текст программы, запрашивающей имя пользователя и создающей файл, в который она будет записывать введённое имя(рис. 4.1).

```

1 %include 'in_out.asm'
2
3 SECTION .data
4 filename db 'final.txt', 0h ; Имя файла
5 msg db 'Как Вас зовут? ', 0h ; Сообщение
6 msg1 db 'Меня зовут ', 0h ;
7
8 SECTION .bss
9 contents resb 255 ; переменная для вводимой строки
10 SECTION .text
11
12 global _start
13
14 _start:
15     ; --- Печать сообщения 'msg'
16     mov eax,msg
17     call sprint
18     ; ---- Запись введенной с клавиатуры строки в 'contents'
19     mov ecx, contents
20     mov edx, 255
21     call sread
22
23     ; --- Создание файла для записи имени
24
25     mov ecx,0777o ; разрешения
26     mov ebx,filename ; имя файла
27     mov eax,8 ; номер сис вызова для создание
28     int 80h
29
30     ; --- Открытие существующего файла ('sys_open')
31
32     mov ecx,2 ; открываем для записи (2)
33     mov ebx,filename
34     mov eax,5
35     int 80h
36     ; --- Запись дескриптора файла в 'esi'
37
38     mov esi,eax
39
40     ; --- Расчет длины введенной строки msg1
41     mov eax,msg1
42     call slen

```

Рис. 4.1: Редактирование файла

Создаю исполняемый файл и запускаю его, ввожу свои имя и фамилию.(рис. 4.2).

```

apkorchagin@dk2n26 ~/work/arch-pc/lab10 $ touch lab10-2.asm
apkorchagin@dk2n26 ~/work/arch-pc/lab10 $ nasm -f elf lab10-2.asm && ld -m elf_i386 -o lab10-2 lab10-2.o
&& ./lab10-2
Как Вас зовут? Алексей Корчагин
apkorchagin@dk2n26 ~/work/arch-pc/lab10 $ ls -l
итого 32

```

Рис. 4.2: Исполняемый файл

С помощью cat проверяю содержимое созданного(рис. 4.3).

```
apkorchagin@dk2n26 ~/work/arch-pc/lab10 $ cat final.txt
Меня зовут Алексей Корчагин
apkorchagin@dk2n26 ~/work/arch-pc/lab10 $ █
```

Рис. 4.3: Проверка содержимого

Код программы:

```
%include 'in_out.asm'

SECTION .data
filename db 'final.txt', 0h ; Имя файла
msg db 'Как Вас зовут? ', 0h ; Сообщение
msg1 db 'Меня зовут ', 0h ;

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

    ; --- Создание файла для записи имени
```

```

mov ecx,0777o ; разрешения
mov ebx,filename ; имя файла
mov eax,8 ; номер сис вызова для создание
int 80h

; --- Открытие существующего файла (`sys_open`)

mov ecx,2 ; открываем для записи (2)
mov ebx,filename
mov eax,5
int 80h

; --- Запись дескриптора файла в `esi`

mov esi,eax

; --- Расчет длины введенной строки msg1
mov eax,msg1
call slen

; --- Записываем в файл `msg1` (`sys_write`)
mov edx, eax
mov ecx, msg1
mov ebx, esi
mov eax, 4
int 80h

; --- Расчет длины введенной строки
mov eax, contents ; в `eax` запишется количество
call slen ; введенных байтов

```

```
        ; --- Записываем в файл `contents` (`sys_write`)
mov     edx, eax
mov     ecx, contents
mov     ebx, esi
mov     eax, 4
int     80h

        ; --- Закрываем файл (`sys_close`)
mov     ebx, esi
mov     eax, 6
int     80h

call    quit
```


5 Выводы

В ходе выполнения лабораторной работы я приобрёл опыт работы с файлами в NASM

Список литературы