

РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

Факультет физико-математических и естественных наук

Кафедра прикладной информатики и теории вероятностей

ОТЧЕТ

ПО ЛАБОРАТОРНОЙ РАБОТЕ № 10

дисциплина: Архитектура компьютера

Студент: Мурылев Иван Валерьевич

Группа: НПИбд-03-25

МОСКВА

2025 г.

Содержание

Цель работы

Теоретическое введение

Выполнение работы

Вывод

Цель работы

Приобретение навыков написания программ для работы с файлами

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. Владелцем файла является его создатель. Для предоставления прав доступа другому пользователю или другой группе командой

chown [ключи] <новый_пользователь>[:новая_группа] <файл>

или

chgrp [ключи] < новая_группа > <файл>

Таблица 10.1. Двоичный, буквенный и восьмеричный способ записи триады прав доступа

Двоичный	Буквенный	Восьмеричный
111	гwx	7
110	гw-	6
101	г-x	5
100	г--	4
011	-wx	3
010	-w-	2
001	--x	1
000	---	0

Полная строка прав доступа в символьном представлении имеет вид:

<права_владельца> <права_группы> <права_остальных>

помощью следующей команды:

```
$ls -l /home/debugger/README
-rwxr-xr-- 1 debugger users 0 Feb 14 19:08 /home/debugger/README
```

В первой колонке показаны текущие права доступа, далее указан владелец файла и группа:

-	rwx	r-x	r-x	1 debugger user
тип	владелец	группа	остальные	
type	owner	group	others	

Основы команды `chmod`:

```
chmod <категория><действие><набор_прав><файл>
```

Возможные значения аргументов команды представлены в таблице 10.2.

Таблица 10.2. Возможные значения аргументов команды `chmod`

Категория	Обозначение	Значение
Принадлежность	u	Владелец
	g	Группа владельца
	o	Прочие пользователи
	a	Все пользователи, то есть «a» эквивалентно «ugo»
Действие	+	Добавить набор прав
	-	Отменить набор прав
	=	Назначить набор прав
Право	r	Право на чтение
	w	Право на запись
	x	Право на исполнение

Работа с файлами средствами Nasm:

Общий алгоритм работы с системными вызовами в Nasm можно представить в следующем виде:

1. Поместить номер системного вызова в регистр EAX;
2. Поместить аргументы системного вызова в регистрах EBX, ECX и EDX;
3. Вызов прерывания (int 80h);
4. Результат обычно возвращается в регистр EAX.

Таблица 10.3. Системные вызовы для обработки файлов

Имя системного вызова	eax	ebx	ecx	edx
sys_read	3	дескриптор файла	адрес в памяти	количество байтов
sys_write	4	дескриптор файла	строка	количество байтов
sys_open	5	имя файла	режим доступа к файлу	права доступа к файлу
sys_close	6	дескриптор файла	—	—
sys_creat	8	имя файла	права доступа к файлу	—
sys_unlink	10	имя файла	—	—
sys_lseek	19	имя файла	значение смещения в байтах	позиция для смещения

Выполнение работы

Лабораторная часть

Так как в структуре репозитория уже создана папка отчета, первый этап пропускается. Переносим in-out.asm в данную директорию.

1. После создаю требуемые файлы:

```

/home/ivmurihlev/Documents/work/study/2025-2026/arch_Evm/arch-evm/labs/lab10/report
[ivmurihlev@personal report]$ touch lab10-1.asm readme-1.txt readme-2.txt
[ivmurihlev@personal report]$ ls
arch-pc--lab10--report.qmd  bib  lab10-1.asm  _quarto.yml  readme-2.txt
_assets  image  Makefile  readme-1.txt  _resources

```

2. Потом переношу Листинг 10.1 в lab10-1.asm и проверяю это командой cat:

```

[ivmurihlev@personal report]$ cat lab10-1.asm
;-----
; Запись в файл строки введенной на запрос
;-----
%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
int 80h
; --- Запись дескриптора файла в `esi`
mov esi, eax
; --- 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

```

Затем создаю исполняемый файл и запускаю его:

```

[ivmurihlev@personal report]$ nasm -f elf lab10-1.asm
[ivmurihlev@personal report]$ ld -m elf_i386 lab10-1.o -o lab10-1
[ivmurihlev@personal report]$ ./lab10-1
Введите строку для записи в файл: wwwwwwwww
[ivmurihlev@personal report]$ cat readme-1.txt
[ivmurihlev@personal report]$ cat readme-2.txt
[ivmurihlev@personal report]$ touch readme.txt
[ivmurihlev@personal report]$ ./lab10-1
Введите строку для записи в файл: 444444444
[ivmurihlev@personal report]$ cat readme.txt
444444444

```

Для корректной работы файла надо было создать файл readme.txt, либо изменить аргумент файла вывода. (Строка 6).

3. Меняю права на исполнение для себя (-u – я владелец ЭВМ), с помощью команды -x (исполнение).

```
[ivmurihlev@personal report]$ chmod -u -x lab10-1
[ivmurihlev@personal report]$ ./lab10-1
bash: ./lab10-1: Permission denied
```

Как видно команда не исполняется из-за запрета.

4. Теперь разрешим исполнение lab10-1 и запретим исполнение lab10-1.asm:

```
[ivmurihlev@personal report]$ chmod +x lab10-1
[ivmurihlev@personal report]$ chmod -x lab10-1.asm
[ivmurihlev@personal report]$ ./lab10-1
Введите строку для записи в файл: eeeeeeeeeee
[ivmurihlev@personal report]$ cat readme.txt
eeeeeeeeeeee
```

Как видно программа работает по причине того что файл lab10-1.asm не исполняется напрямую. Корректность проверяется с помощью cat.

5. Мой вариант седьмой:

7 rw- rwx rw- 101 111 111

Выполняем в терминале команды следующие для readme-1.txt и проверяем через команду `ls -l` :

```
[ivmurihlev@personal report]$ chmod u=rw-,g=rwx,o=rw- readme-1.txt
[ivmurihlev@personal report]$ ls -l readme-1.txt
-rw-rwxrw- 1 ivmurihlev ivmurihlev 0 дек 14 15:36 readme-1.txt
```

Тоже самое для readme-2.txt только переведем из двоичного в восьмиричную:

```
[ivmurihlev@personal report]$ ls -l readme-2.txt
-r-xrwxrwx 1 ivmurihlev ivmurihlev 0 дек 14 15:36 readme-2.txt
```

Самостоятельная часть

Создаем файл lab10-2.asm :

```
[ivmurihlev@personal report]$ touch lab10-2.asm
```

После его заполнения и создания исполняемого файла запускаем:

```
[ivmurihlev@personal report]$ nasm -f elf lab10-2.asm
[ivmurihlev@personal report]$ ld -m elf_i386 lab10-2.o -o lab10-2
[ivmurihlev@personal report]$ ./lab10-2
Фамилия и имя: Мурылев Иван
Введите строку для записи в файл: Я старался.Честно
[ivmurihlev@personal report]$ cat name.txt
Мурылев Иван
Я старался.Честно
```

Как видно программа работает верно и все записывает верно .Проверка через *cat name.txt* .

Вывод

Особенность распределения прав в ОС на ядре Lnx делает его удобным, когда его надо использовать на не персональных компьютерах. Механизм прав допуска является важной частью данного вида операционной системы. Также были освоены основы работы с файлами на языке Assambler.