МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
«МИРЭА – РОССИЙСКИЙ ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ»
РТУ МИРЭА

Институт ИКБ

Специальность (направление): 09.03.02 (информационные системы и технологии)

Кафедра: КБ-3 «Разработка программных решений и системного программирования»

Дисциплина: “Безопасность операционных систем”

Практическая работа
на тему:
Практическое занятие № 7

Студент: _____ 13.12.2024 Крашенинников М.В.
подпись *Дата* *инициалы и фамилия*

Группа: БСБО-16-23 Шифр: 23Б0107

Преподаватель: _____ 13.12.2024 Иванова И. А.
подпись *дата* *инициалы и фамилия*

Москва 2024 г.

Ход работы

До этого создал пользователя guest и задал ему пароль

```
$ ls
dir1
$ ls -l dir1
total 0
-rw-r--r-- 1 guest guest  0 Dec 12 11:57 dummy_file
-rw-rw-r-- 1 guest guest  5 Dec 12 12:10 file1
-rw-rw-r-- 1 guest guest 14 Dec 12 12:04 real_file
-r-x------ 1 guest guest 31 Dec 12 12:07 test.sh
$ stat dir1
  File: dir1
  Size: 4096            Blocks: 0          IO Block: 4096   directory
Device: 2h/2d   Inode: 39406496739777619  Links: 1
Access: (0700/drwx------)  Uid: ( 1001/   guest)   Gid: ( 1001/   guest)
Access: 2024-12-13 23:14:36.044867200 +0300
Modify: 2024-12-12 12:10:06.527283900 +0300
Change: 2024-12-12 12:10:06.527283900 +0300
 Birth: -
$ nano dir1/test.sh
$ chmod 500 dir1/test.sh
$ chmod 000 dir1
$ ls -l
total 0
d--------- 1 guest guest 4096 Dec 13 23:15 dir1
$ echo "test">dir1/file1
-sh: 8: cannot create dir1/file1: Permission denied
$ sudo echo "test">dir1/file1
-sh: 9: cannot create dir1/file1: Permission denied
$ mv dir1/file1 dir1/file2
mv: failed to access 'dir1/file2': Permission denied
$ cat dir1/test.sh
cat: dir1/test.sh: Permission denied
$ chmod 700 dir1
$ ls -l
total 0
drwx------ 1 guest guest 4096 Dec 13 23:15 dir1
$ echo "test">dir1/file1
$ cat dir1/test.sh
#! /bin/bash
echo "praktika 7"
$ mv dir1/file1 dir1/file2
$ rm dir1/file2
```

Задание для самостоятельной работы:

Таблица 1. Установленные права и разрешённые действия

Директории	Файлы	Удаление файла	Запись в файл	Чтение файла	Просмотр файлов	Переименование файла	Исполнение файла
0000/	0000/	-	-	-	-	-	-
0100/	0000/	-	-	-	+	-	-
0200/	0000/	-	-	-	-	-	-
0300/	0000/	-	-	-	+	-	-
0400/	0000/	-	-	+	+	-	-
0500/	0000/	-	-	+	+	-	+
0600/	0000/	-	+	+	+	-	+
0700/	0700/	+	+	+	+	+	+

Таблица 2. Минимальные права для совершения операций

Операция	Минимальные права на директорию	Минимальные права на файл (поддиректорию)
Создание файла	`w`	-
Удаление файла	`w`	-
Чтение файла	`x`	`r`
Запись в файл	`x`	`w`
Переименование файла	`w`	-
Создание поддиректории	`w`	-
Удаление поддиректории	`w`	-

Ответы на вопросы

1. Конфигурация подсистемы защиты в ОС

Подсистема защиты обеспечивает конфиденциальность, целостность и доступность данных в системе. Она включает механизмы разграничения прав доступа, шифрование данных, системы аудита и контроля, а также настройку политик безопасности. Конфигурация осуществляется через файлы конфигурации, интерфейсы ОС или специальные инструменты администрирования, такие как SELinux или AppArmor.

2. Механизм идентификации пользователей в ОС

Идентификация пользователей — это процесс определения уникальности пользователя. В ОС каждый пользователь имеет уникальный идентификатор (например, UID в Linux), который связывается с его учетной записью.

3. Механизм аутентификации пользователей в ОС

Аутентификация — это подтверждение личности пользователя. Она может быть реализована через пароли, ключи, биометрические данные или токены. Пример: ввод пароля при входе в систему или использование команд sudo с подтверждением через аутентификацию.

4. Основные механизмы защиты в ОС

1) Разграничение доступа (ACL, права на файлы и каталоги).
2) Шифрование данных (дисковое шифрование, SSL/TLS).
3) Аудит и мониторинг (журналы событий).
4) Изоляция процессов (контейнеризация, виртуализация).

5. Классификация угроз

1) Физические (кража оборудования).
2) Программные (вирусы, трояны, руткиты).
3) Сетевые (DoS-атаки, перехват данных).
4) Социальная инженерия (фишинг, взлом через доверие)

6. Авторизация. Разграничение доступа к объектам ОС

Авторизация — это проверка прав пользователя на выполнение операций с ресурсами. Механизмы:

1) POSIX-права: чтение, запись, выполнение.
2) ACL (Access Control Lists): гибкие списки прав доступа.
3) SELinux: мандатное управление доступом.

7. Аудит системы защиты

Аудит фиксирует действия пользователей и процессов для выявления попыток взлома, анализа инцидентов или оценки текущего уровня безопасности. Пример: команды journalctl и просмотр /var/log/auth.log в Linux.

8. Системы защиты программного обеспечения

Системы защиты ПО предотвращают копирование, модификацию или несанкционированное использование. Методы:

1) Лицензионные ключи.
2) Шифрование кода.
3) Программы-защитники (антивирусы, фаерволы).

9. Атаки на операционные системы

Популярные виды атак:

1) Эксплойты (использование уязвимостей).
2) Атаки на пароли (перебор, фишинг).
3) Руткиты (маскировка вредоносного ПО).
4) Межпроцессные атаки (например, переполнение буфера).

10. Защищенные операционные системы

Защищённые ОС имеют встроенные механизмы безопасности, например:

1) Windows с BitLocker.
2) Linux с SELinux/AppArmor.
3) Qubes OS (использует изоляцию через виртуальные машины).

11. Получение данных авторизации и другой ключевой информации

Методы:

1) Сниффинг (перехват данных в сети).
2) Кража через вредоносные программы.
3) Социальная инженерия.
4) Брутфорс.

12. Восстановление удаленных данных (сборка мусора)

Удалённые данные можно восстановить, если они не были перезаписаны. Используются программы вроде TestDisk, photorec. Также применяется анализ мусора (лог-файлы, временные данные).

13. Поиск и сбор информации

Методы:

1) Легальные: лог-файлы, системные утилиты (например, ps, top).
2) Неавторизованные: сканирование портов, исследование метаданных.

14. Аппаратная и биометрическая аутентификация

1) Аппаратная: использование USB-токенов, смарт-карт.
2) Биометрическая: сканирование отпечатков пальцев, сетчатки глаза, распознавание лиц.

15. Модели управления доступом

1) Discretionary Access Control (DAC): доступ определяется владельцем объекта.
2) Mandatory Access Control (MAC): доступ определяется политиками безопасности (например, SELinux).
3) Role-Based Access Control (RBAC): доступ предоставляется в зависимости от роли пользователя.