Федеральное осударственное автономное образовательное учреждение
высшего образования «Самарский национальный исследовательский университет
имени академика С.П. Королева»
(Самарский университет)

Лабораторная работа №4
Безопасность в UNIX

Самара, 2021

Содержание

1. ОСНОВНЫЕ ОПРЕДЕЛЕНИЯ

1. **Учетная запись пользователя** – уникальное в пределах системы имя, с которым связан набор параметров (в т.ч. набор привилегий).

2. **Идентификатор пользователя (UID)** – уникальное число, идентифицирующее пользователя в системе.

3. **Привилегии** – разрешение на выполнение определенных действий.

4. **Суперпользователь** – пользователь, обладающий всеми возможными привилегиями.

5. **Пользователь root** – суперпользователь, имеющийся в любой системе UNIX, имеющий UID = 0.

6. **Реальный идентификатор пользователя (rUID)** – идентификатор пользователя, запустившего программу.

7. **Эффективный идентификатор пользователя (eUID)** – идентификатор пользователя, который используется ОС при проверке привилегий во время выполнения программы.

8. **Бит set-user-id** – бит прав доступа, который инструктирует ОС при запуске исполняемого файла с установленным set-user-id запускать выполнение соответствующей программы от имени владельца файла, а не от имени пользователя, запустившего файл (т.е. eUID != sUID).

2. ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

2.1. Бит set-user-id

Бит set-user-id используется многими системными приложениями для доступа к защищенным системным ресурсом. Поскольку владельцем системных приложений является root, данные приложения имеют получают полный доступ к системе.

Изменить владельца файла на root и установить бит set-user-id можно следующими командами:

```
sudo chown root:root FILE
sudo chmod u+s FILE
```

2.2. Изменение ID пользователя

В UNIX с каждой программой связано как минимум 2 идентификатора пользователя – реальный идентификатор пользователя и эффективный идентификатор пользователя.

Получить идентификаторы пользователя можно следующими вызовами:

```
#include <unistd.h>
#include <sys/types.h>

uid_t getuid();  // реальный UID
uid_t geteuid(); // эффективный UID
```

Изменить эффективный UID можно вызовом setuid:

```
#include <unistd.h>
#include <sys/types.h>

int setuid(uid_t uid);
```

Обычный пользователь имеет право изменить эффективный идентификатор либо на реальный, либо на сохраненный (см. лекцию 2) идентификатор. Т.е. наиболее простой является конструкция `setuid(getuid()).`

Исключением является суперпользователь – он имеет право изменить свой идентификатор на идентификатор любого пользователя, **НО** при этом изменятся все идентификаторы (и эффективный, и реальный). То есть, как только программа отказывается от прав суперпользователя, она больше не может получить их обратно, так как реальный идентификатор изменился (если, конечно, реальный идентификатор не является также идентификатором суперпользователя – то есть, если программу запустил сам суперпользователь).

2.3. Проверка привилегий

В UNIX проверка привилегий происходит в момент получения ресурса: открытия файла, присоединения сокета к конечной точке и пр.

После того, как ресурс получен, дальнейшие операции не проверяются (вернее, проверяются только на соответствие пользовательским или системным ограничениям на ресурсы).

Если программа запускается от имени суперпользователя, то она имеет максимальнеый набор привилегий, что порождает существенную угрозу безопасности, если программа будет взломана или еще каким-то образом вынужден выполнять вредоносные действия.

Для минимизации временного окна уязвимости, применяется следующая схема.

- программа запускается с правами суперпользователя;
- программа открывает необходимые ресурсы;
- программа выполняет `setuid(getuid())`, тем самым отказываясь от прав суперпользователя.

```
int fd = open("/proc/1/maps", O_RDONLY); /*open file*/

setuid(getuid()); /*drop root rights*/

/*use file*/

close(fd);
```

2.4. Системные файлы с информацией о пользователях.

Публичная (доступная всем) информация о пользователях расположена в файле /etc/passwd
Одна строка соответствует одному пользователю. Формат строки:

(имя):(x):(uid):(gid):(доп.инф.):(каталог):(оболочка)

Закрытая информация о пользователях хранится в файле /etc/shadow
Одна строка соответствует одному пользователю. Формат строки:

(имя):(пароль):<др. атрибуты>

Закрытая информация о группах пользователей приведена в файле /etc/gshadow.
Одна строка соответствует одной группе. Формат строки:

(имя группы):(пароль группы):(администраторы):(пользователи)

Во избежание повреждения системных файлов, настоятельно рекомендуется открывать их только на чтение.

Для облегчения задачи разрешается (только в этой л/р!) для чтения файлов использовать std::ifstream.

3. ЗАДАНИЕ НА ЛАБОРАТОРНУЮ РАБОТУ

Написать простую программу, выводящую информацию обо всех пользователях в системе.
Как минимум, выводить:
- UID;
- имя пользователя;
- домашний каталог;
- хэш пароля;
- список групп (если пользователь – админ группы, отметьте это любым образом).

При этом любой пользователь должен иметь возможность запустить программу.

Помните, что данная программа пишется в целях обучения и в реальных системах использоваться не должна.