**ОБЩАЯ ТЕОРИЯ**

**Что такое SUID-бит?**

**SUID (Set User ID)** — это специальный флаг в операционных системах, таких как Linux и macOS, который может быть установлен на исполнимые файлы. Когда SUID-бит установлен для программы, программа будет выполняться с правами владельца файла, а не с правами того пользователя, который её запускает. То есть, даже если обычный пользователь запускает программу, она будет выполняться с правами суперпользователя (root) или другого владельца файла.

**Пример использования SUID**

Предположим, что у нас есть программа, которая читает файл, доступ к которому есть только у владельца файла (например, root), но этот файл нужно прочитать пользователю, который не имеет доступа к нему напрямую. Если на этой программе установлен SUID-бит, то программа будет запускаться с правами владельца, а не с правами того пользователя, который её запускает. Это позволяет пользователям выполнять действия с привилегиями владельца программы.

**Как работает SUID?**

Когда файл имеет установленный SUID-бит, операционная система изменяет процесс таким образом, чтобы во время выполнения программы использовались права владельца файла. Процесс, который запустил программу, всё равно остаётся тем же пользователем, но при этом система "обманывает" её, давая ей права владельца файла.

**Как установить SUID?**

Чтобы установить SUID-бит на файл, используется команда chmod. Например:

chmod u+s example\_program

После этого флаг s появляется в правой части прав доступа к файлу, как это видно в выводе команды ls -l:

-rwsr-xr-x 1 root root 12345 Apr 10 14:39 example\_program

**Что такое UID?**

**UID (User Identifier)** — это уникальный идентификатор пользователя в операционной системе, который используется для управления доступом к системным ресурсам. Каждый пользователь в системе имеет свой UID.

**Реальный UID (Real UID)** — это UID пользователя, который запустил процесс. Это может быть обычный пользователь или root.

**Эффективный UID (Effective UID)** — это UID, который операционная система использует для проверки прав доступа при выполнении операций в процессе. Если на программу установлен SUID-бит, то эффективный UID будет равен UID владельца программы, а не пользователя, который её запустил.

**Зачем нужен SUID-бит?**

- Права доступа к ограниченным ресурсам: Когда обычный пользователь должен выполнить программу, которая требует прав суперпользователя для доступа к определённым файлам или ресурсам (например, запуск программы, которая изменяет настройки системы).

- Упрощение доступа к необходимым действиям: Например, программы, такие как passwd, используют SUID-бит, чтобы обычный пользователь мог изменять свой пароль (хотя для этого нужно редактировать системные файлы, доступные только суперпользователю).

- Обеспечение безопасности: SUID позволяет избежать передачи прав суперпользователя целиком (через sudo), что снижает риски случайных ошибок.

**Потенциальные проблемы и опасности SUID**

- Уязвимости в программе: Если программа с SUID-битом содержит ошибку (например, ошибку в безопасности), злоумышленник может воспользоваться этой уязвимостью для получения прав суперпользователя. Это особенно опасно, если программа выполняет критичные операции с системой.

- Несанкционированный доступ: Если кто-то получит доступ к SUID-программе, он может использовать её для выполнения операций, требующих прав суперпользователя, что может привести к изменению важных системных файлов или данных.

- Трудности с контролем: В системе может быть много программ с установленным SUID-битом, что усложняет контроль над безопасностью. Пользователи могут случайно или намеренно запустить вредоносные программы, получив несанкционированный доступ к системе.  
  
**Почему это может быть плохо?**

SUID-бит может быть небезопасным, если:

- Программа не проверяет корректно пользовательский ввод или выполняет операции, которые могут быть использованы для повышения привилегий.

- Программа с SUID имеет уязвимости или ошибки безопасности, которые могут быть использованы злоумышленниками для получения привилегий root.

- Программы с SUID-битом обычно не проходят тщательную проверку на безопасность, что делает их потенциально уязвимыми.

**Как минимизировать риски?**

- Ограничьте количество SUID-программ: Используйте SUID-бит только там, где это необходимо, и регулярно проверяйте систему на наличие ненужных SUID-файлов.

- Обновляйте программы: Обновляйте программы, установленные с SUID-битом, чтобы гарантировать, что они не содержат уязвимостей.

**Заключение**

SUID-бит — мощный инструмент для управления правами доступа в системе. Он даёт возможность программам исполняться с правами владельца файла (например, root), что может быть полезно для выполнения операций, требующих повышенных привилегий. Однако неправильное использование или наличие уязвимостей в программах с установленным SUID-битом может привести к серьёзным угрозам безопасности. Поэтому важно тщательно контролировать и ограничивать использование SUID.  
  
**Как работает SUID изнутри?**

Когда на файл устанавливается SUID (Set User ID), операционная система Linux (или macOS) изменяет способ работы с этим файлом таким образом, что при его запуске процесс будет использовать права владельца файла (а не права пользователя, который запустил программу).

**1. Что происходит при запуске программы с SUID-битом?**

Предположим, у нас есть программа с установленным SUID-битом. Когда пользователь запускает эту программу, операционная система начинает следующее:

- Запуск процесса: Процесс создается так, как если бы его запустил пользователь, который инициировал запуск программы. Однако важно, что этот процесс теперь работает с правами владельца файла, а не с правами текущего пользователя.

- Эффективный UID: Когда система запускает процесс, она устанавливает его эффективный UID в UID владельца программы. Это означает, что процесс будет действовать как владелец программы (например, root), даже если её запустил обычный пользователь. Эффективный UID определяет права доступа к системным ресурсам (файлы, устройства, память и т. д.).

- Реальный UID: При этом реальный UID (который указывает на того пользователя, который на самом деле запустил программу) остаётся прежним и не меняется. Реальный UID используется, чтобы система знала, кто инициировал выполнение процесса. Однако для выполнения операций в контексте системы будет использоваться эффективный UID.

- Контроль доступа: Когда программа пытается получить доступ к системным ресурсам (например, открыть файл, изменить его или выполнить команду), операционная система проверяет права доступа через эффективный UID. В случае с программой, где установлен SUID-бит, это будет UID владельца программы.

**2. Как это работает в операционной системе?**

Когда процесс запускается с SUID-битом:

- Операционная система загружает исполнимый файл в память.

- Операционная система изменяет эффективный UID процесса перед его запуском, устанавливая его равным владельцу файла, если на файле установлен SUID-бит.

- Когда процесс пытается получить доступ к файлам или ресурсам, система проверяет доступ через эффективный UID (и только в этом случае будут проверяться права доступа).

-Реальный UID остаётся неизменным и используется в основном для учёта того, кто запустил программу.

**ЗАДАНИЕ**

1. Использование SUID-бита для доступа к файлам:

a. Создайте файл с правами “чтение только владельцу”.

b. Напишите программу, которая выводит:

i. содержимое этого файла.

ii. реальный и эффективный идентификаторы пользователя

c. Запустите программу из-под своего пользователя и из-под чужого.

d. Объясните результат.

e. Установите SUID-бит.

f. Запустите программу из-под своего пользователя и из-под чужого.

g. Объясните результат.

#include <stdio.h>

#include <unistd.h>

#include <stdlib.h>

int main() {

FILE \*file = fopen("secret.txt", "r");

if (!file) {

perror("Ошибка при открытии файла");

return 1;

}

char buffer[256];

fgets(buffer, sizeof(buffer), file);

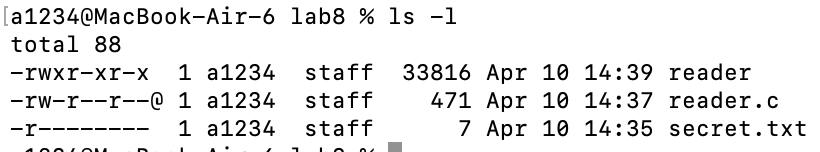
fclose(file);

printf("Содержимое файла: %s\n", buffer);

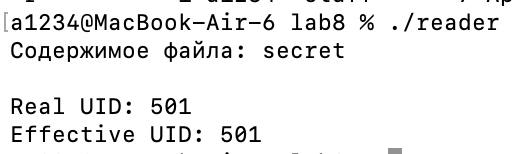
printf("Real UID: %d\n", getuid());

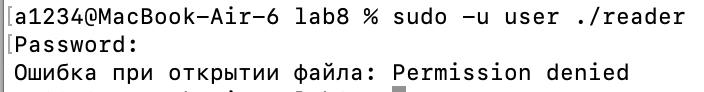
printf("Effective UID: %d\n", geteuid());

return 0;

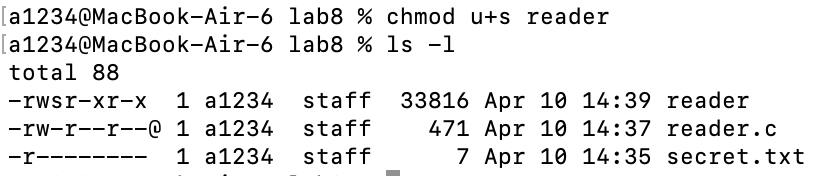
}  
  


**БЕЗ SUID-БИТА:**

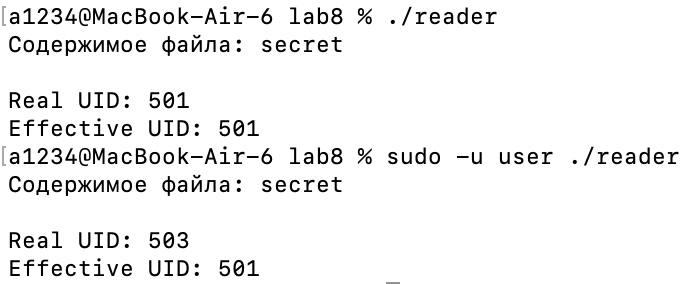




**УСТАНАВЛИВАЕМ SUID-БИТ:**

****

**С SUID-БИТОМ:**

****