МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ

РОССИЙСКОЙ ФЕДЕРАЦИИ

ФГБОУ ВО «НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ «МЭИ»

**Факультет/институт** «Инженерно-экономический институт»

                                                  (название факультета/института)

**Кафедра** «Безопасности и информационных технологий»

            (название кафедры)

**ПРАКТИЧЕСКАЯ РАБОТА № 9**

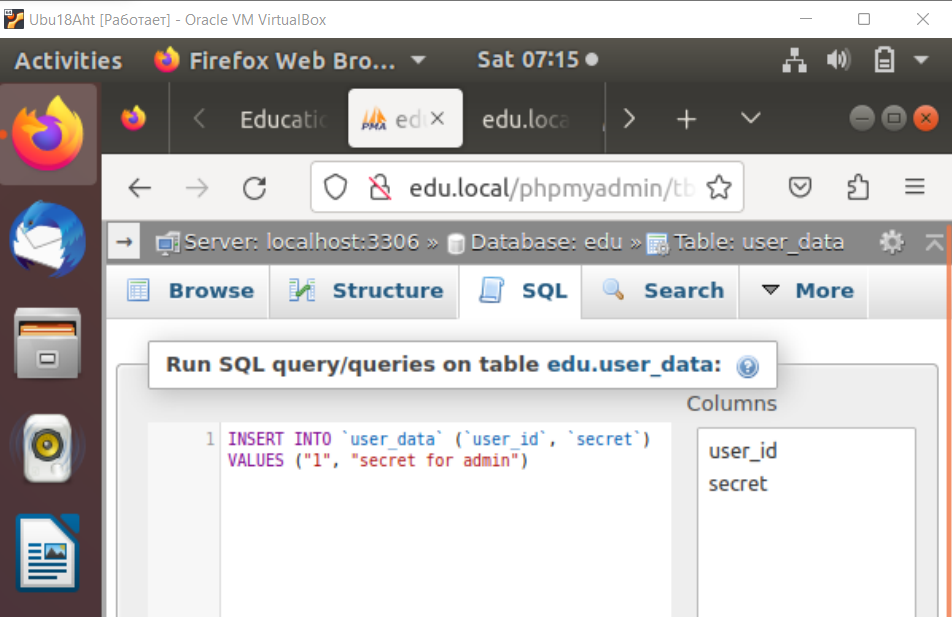
**Дисциплина:**Технологии защиты информационных систем от кибератак  
**Тема:** Предотвращение CSRF

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Выполнил студент | | | 3 | | | курса | | |
| очной формы обучения | | | | | | |  |  |
| Группа №: | ИЭ-41-20 | | | | | | |  |
| ФИО: | Ахтямов В.И. | | | | | | |  |
| Проверил |  | | | | | | |  |
| ФИО: | Шарафутдинов Р.Ф. | | | | | | |  |
| дата проверки: | « |  | | » |  | | 2023г. |  |
| Оценка: |  | | | | | | |  |

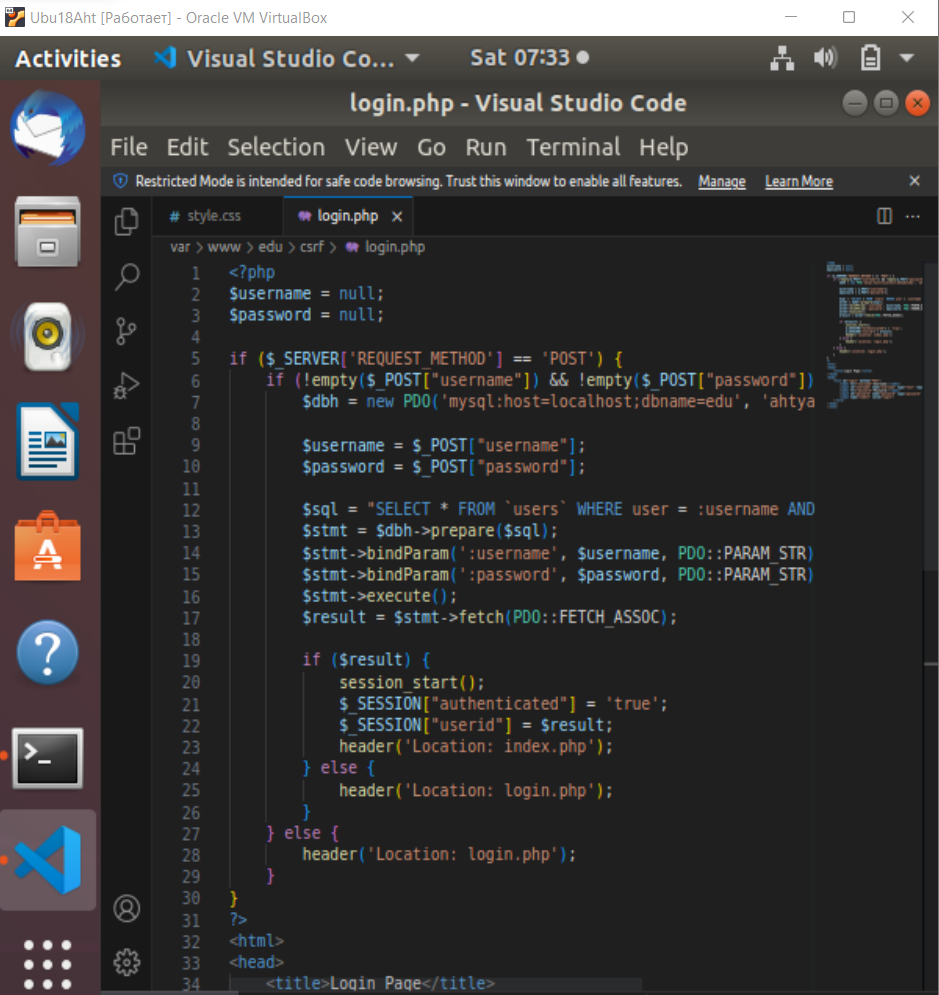
Москва, 2023

Создать таблицу users в базе данных

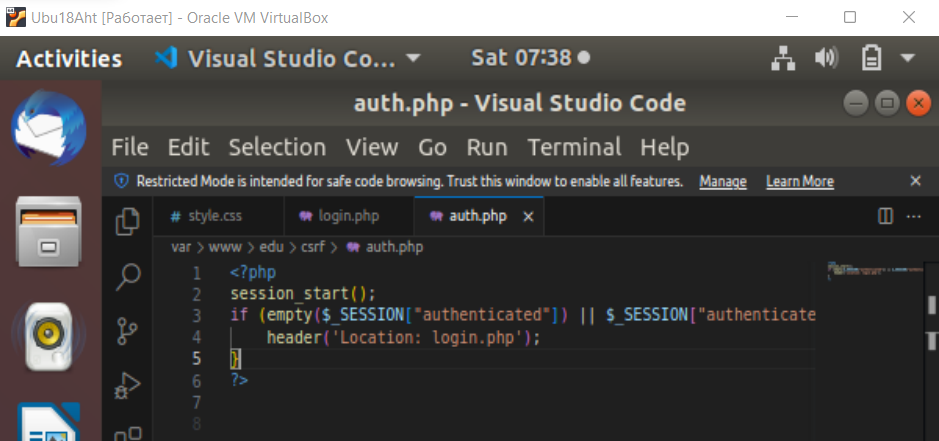
Создать таблицу user\_data в базе данных



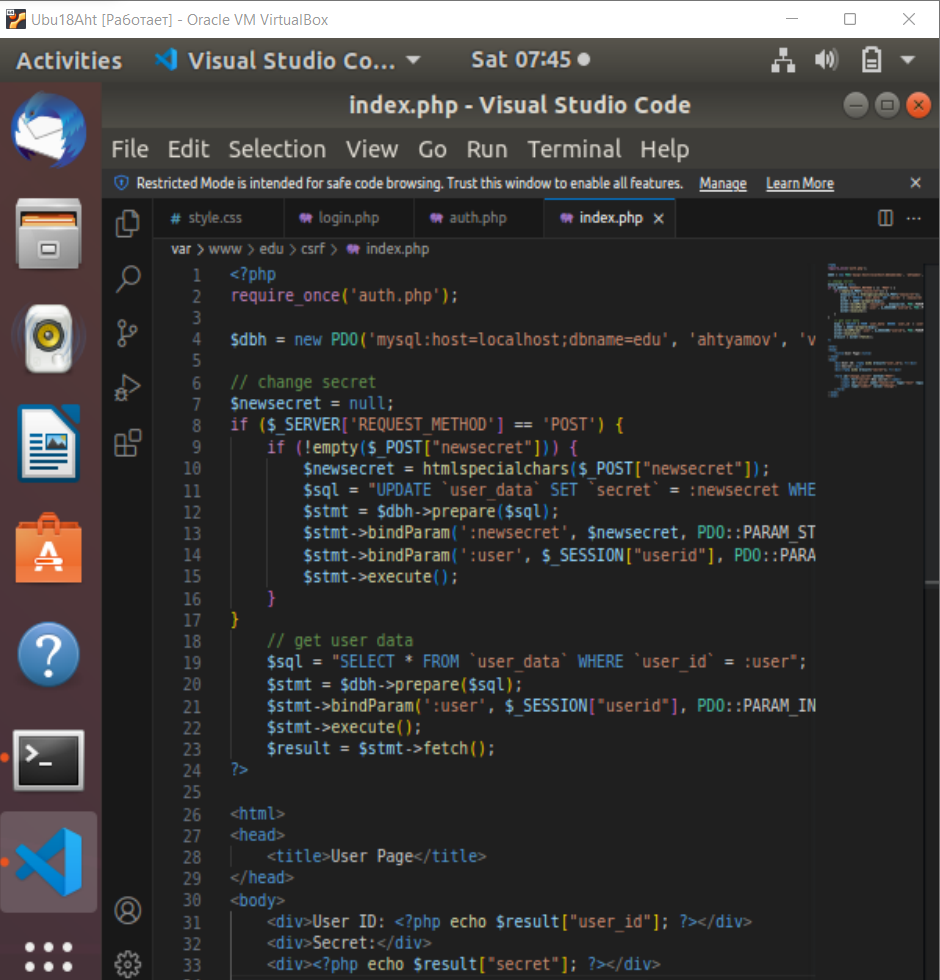
Создать страницу авторизации login.php



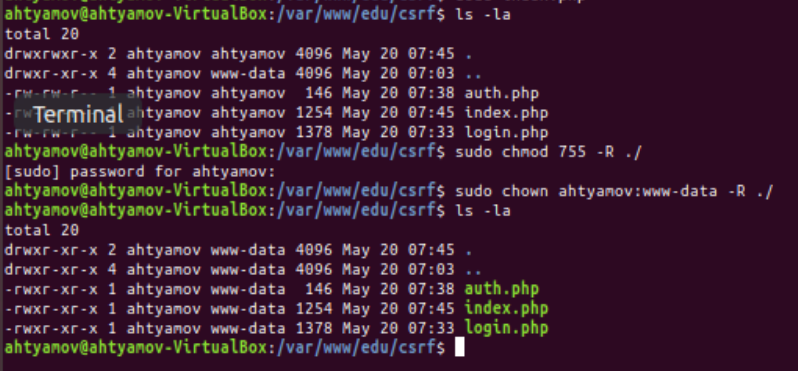
Создать auth.php



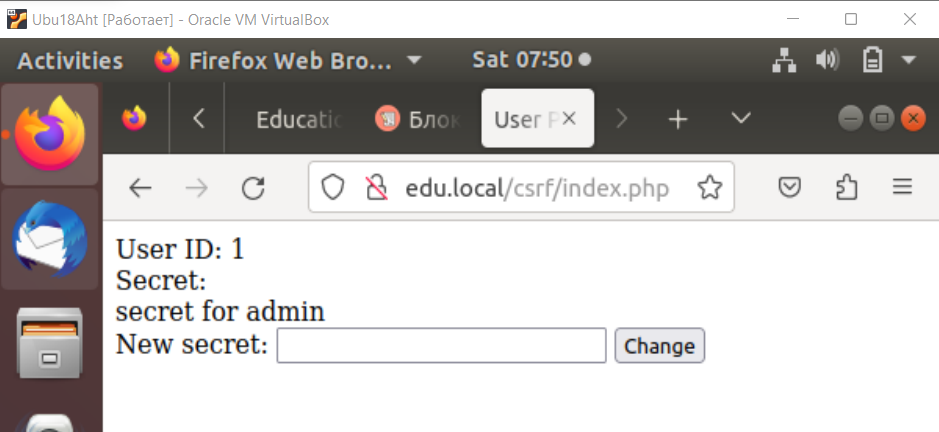
Создать index.php



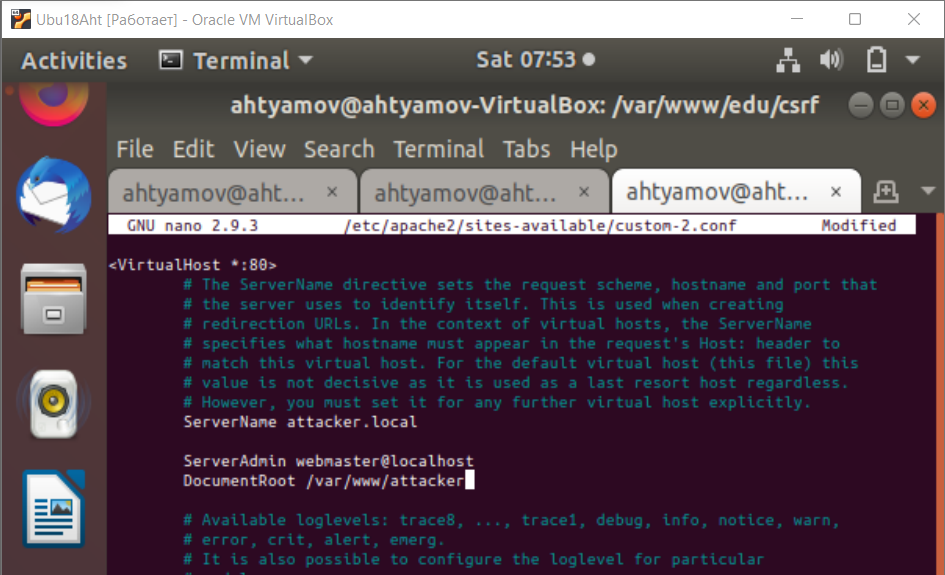
Проверить работоспособность



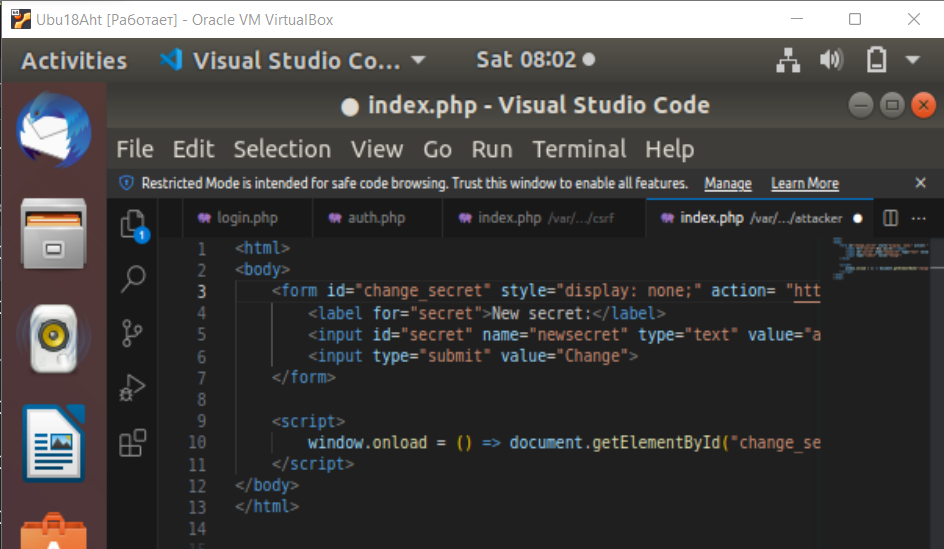
И ура, наконец-то что-то получилось:



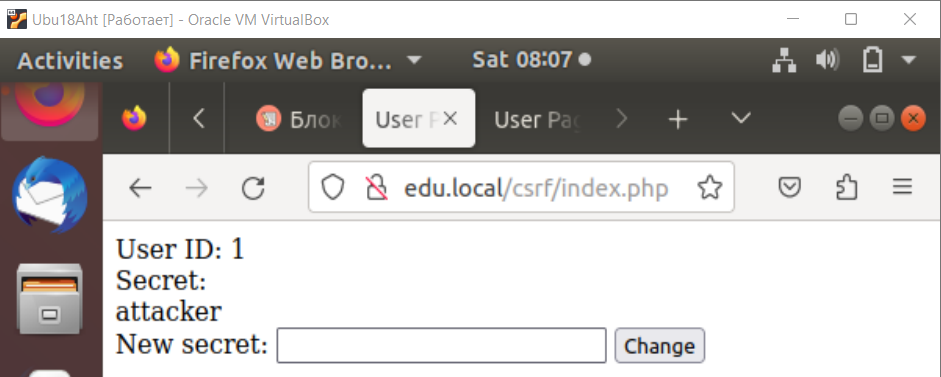
Создать сайт злоумышленника



Создать файлик index.php в /var/www/attacker



И ура, тоже сработало



Наш код уязвим для атаки CSRF (Cross-Site Request Forgery), что означает, что мы не осуществляем проверку подлинности POST-запросов на сервере. Это позволяет злоумышленникам подделывать запросы на изменение секрета пользователя. Давайте рассмотрим это подробнее.

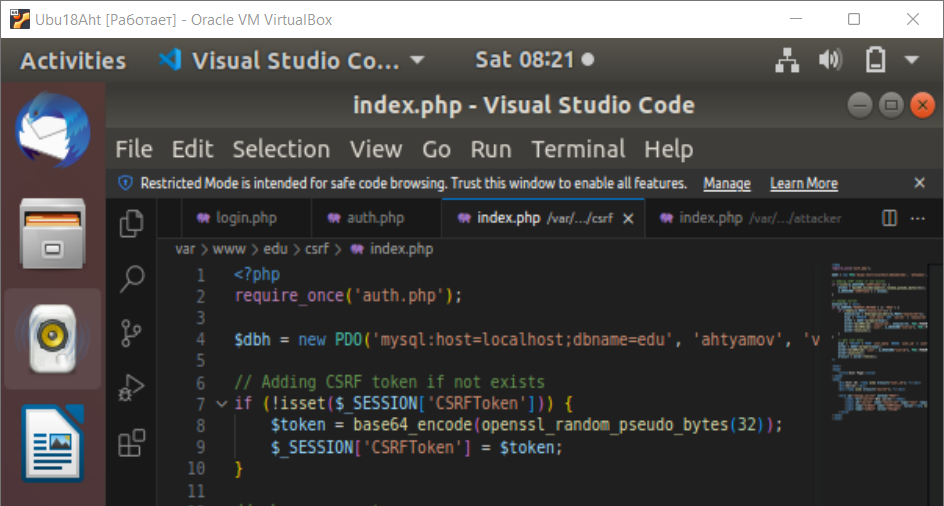
Файл attacker.php представляет собой веб-страницу, которая автоматически отправляет POST-запрос на <http://edu.local/index.php> при загрузке. В этом запросе передается новое значение для "секрета" ("attacker").

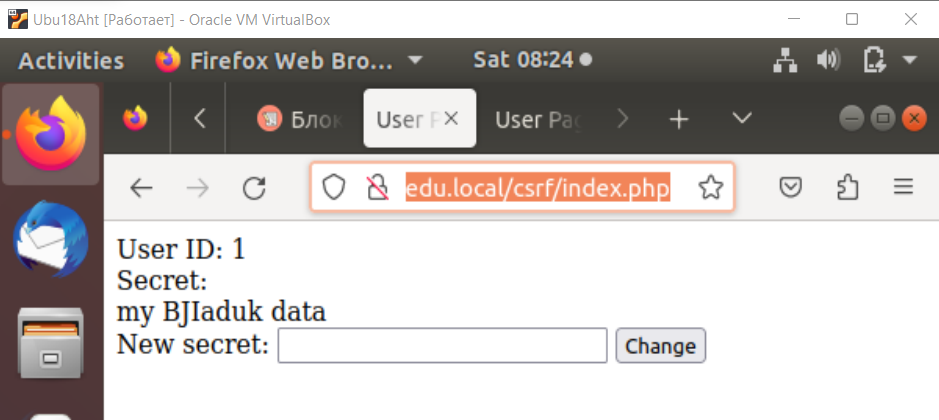
Из-за отсутствия проверки подлинности POST-запросов в нашем коде index.php, сервер безусловно принимает это новое значение и обновляет секрет пользователя в базе данных. В этом и заключается уязвимость для атаки CSRF.

Если бы у нас была защита от CSRF, сервер мог бы проверять каждый входящий POST-запрос на наличие специального CSRF-токена. Этот токен был бы предварительно выдан сервером и связан с пользовательской сессией. При передаче формы для изменения секрета, токен также передавался бы, и сервер проверял бы его при обработке POST-запроса. Если токен отсутствовал или не соответствовал выданному сервером, запрос был бы отклонен.

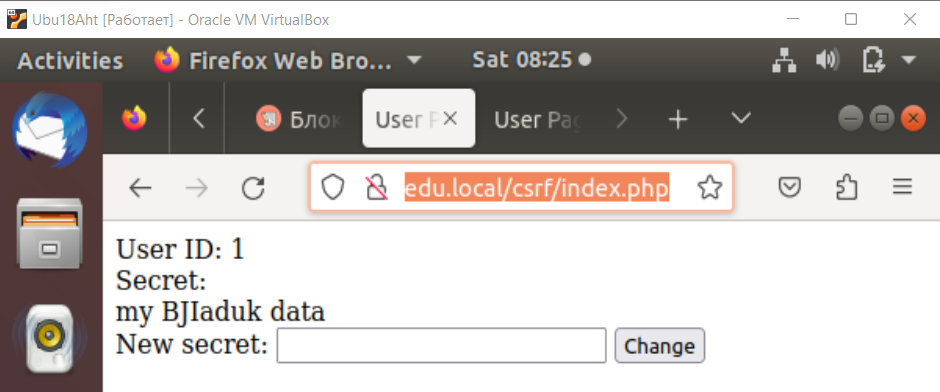
Нашей главной задачей будет внедрение системы CSRF-токенов в наш код, чтобы обеспечить безопасность от атак CSRF.

Изменим файл /var/www/edu/csrf/index.php





И теперь атака не работает



Вывод:

В процессе выполнения практической работы мы изучили уязвимость, известную как CSRF (Cross-Site Request Forgery), которая позволяет злоумышленникам выполнять нежелательные действия от имени аутентифицированного пользователя, не его ведома. Мы продемонстрировали, как злоумышленник может использовать эту уязвимость для изменения "секрета" пользователя на вредоносной веб-странице, которую посещает пользователь. Это стало возможным из-за отсутствия проверки подлинности POST-запросов на сервере. Для устранения этой уязвимости мы применили защитную технику, называемую CSRF-токены.