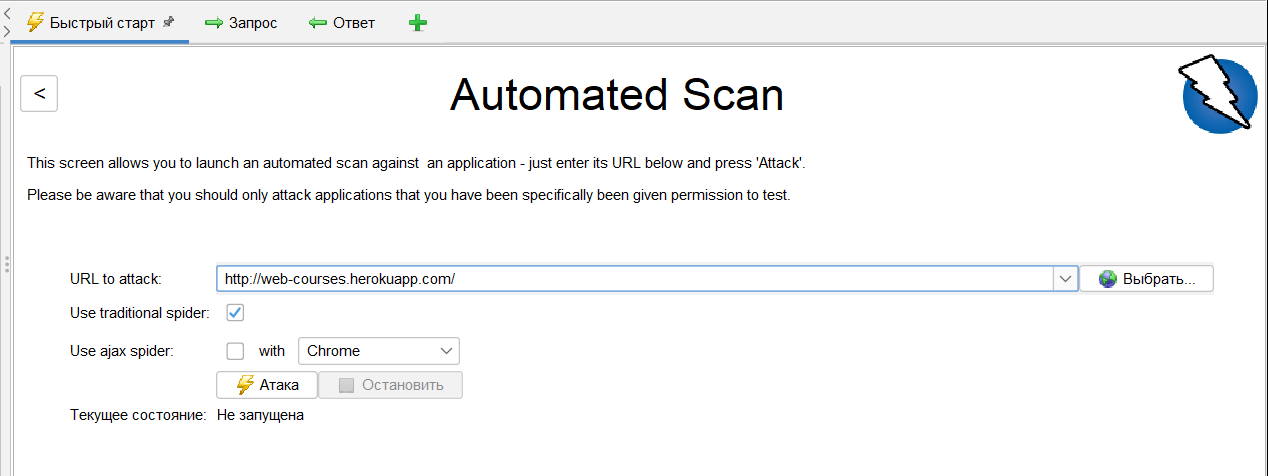
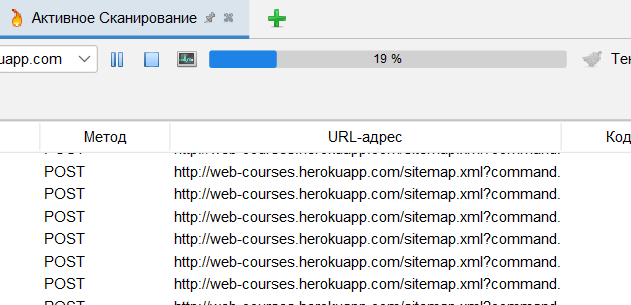
œАсташко Полина, 4 курс 12 группа

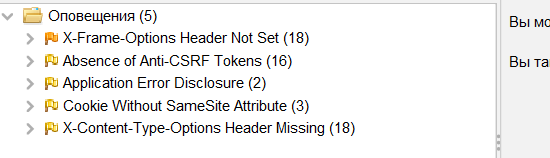
КБРС лаб. №5

В рамках выполнения лабораторной работы было установлено приложение ZAP OWASP. В качестве примера для тестирования был выбран веб-сайт, представляющий собой систему создания учебных факультативов, а также записи на них (аналог Coursera). Веб-сайт развернут на облачной платформе Heroku.





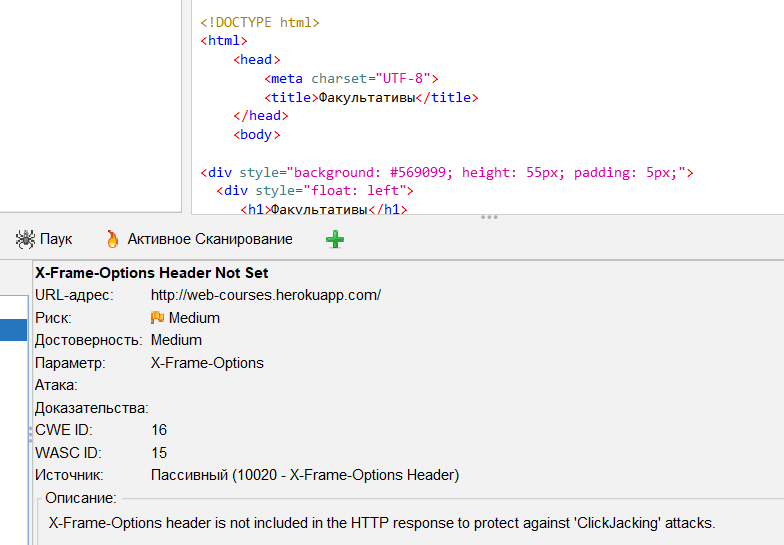
В ходе тестирования было выявлено 5 предупреждений: 1 типа Medium и 4 типа Low.



1. X-Frame-Options Header Not Set

Заголовок X-Frame-Options не включается в HTTP-ответ для защиты от атак ClickJacking.

Может быть исправлено путем обновления CORS политики.



1. Absence of Anti-CSRF Tokens

В форме отправки HTML токенов Anti-CSRF не обнаружено.

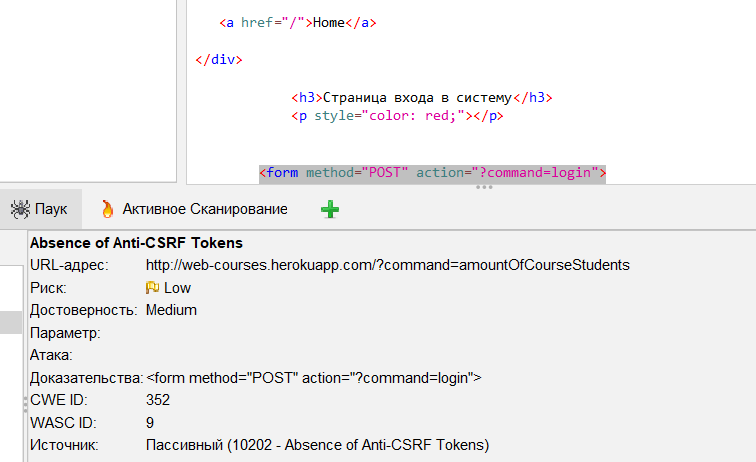
Подделка межсайтового запроса - это атака, которая включает в себя принуждение жертвы к отправке HTTP-запроса в целевой пункт назначения без ее ведома или намерения, чтобы выполнить действие в качестве жертвы. Основная причина - функциональность приложения, использующая предсказуемые действия URL / формы повторяющимся образом. Суть атаки заключается в том, что CSRF использует доверие, которое веб-сайт имеет к пользователю. Напротив, межсайтовый скриптинг (XSS) использует доверие пользователя к веб-сайту. Как и XSS, CSRF-атаки не обязательно являются межсайтовыми, но могут быть. Подделка межсайтовых запросов также известна как CSRF, XSRF, атака в один клик, сессионная атака, сбитый с толку помощник и морской серфинг.

CSRF-атаки эффективны в ряде ситуаций, в том числе:

* У жертвы есть активный сеанс на целевом сайте.
* Жертва аутентифицируется через HTTP-аутентификацию на целевом сайте.
* Жертва находится в той же локальной сети, что и целевой сайт.

CSRF в основном использовался для выполнения действия против целевого сайта с использованием привилегий жертвы, но недавно были обнаружены методы раскрытия информации путем получения доступа к ответу. Риск раскрытия информации резко возрастает, когда целевой сайт уязвим для XSS, потому что XSS может использоваться в качестве платформы для CSRF, позволяя атаке действовать в рамках политики одного и того же происхождения.

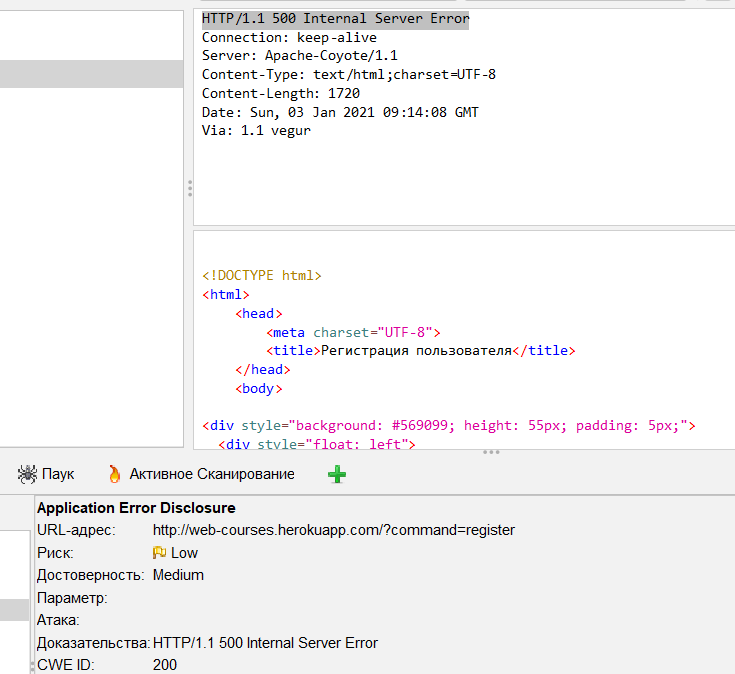
Может быть исправлено путем добавления библиотеки anti-CRSF.



1. Application Error Disclosure

Эта страница содержит сообщение об ошибке / предупреждении, которое может раскрывать конфиденциальную информацию, такую как расположение файла, вызвавшего необработанное исключение. Эта информация может быть использована для дальнейших атак на веб-приложение. Предупреждение может быть ложным, если сообщение об ошибке находится на странице документации.

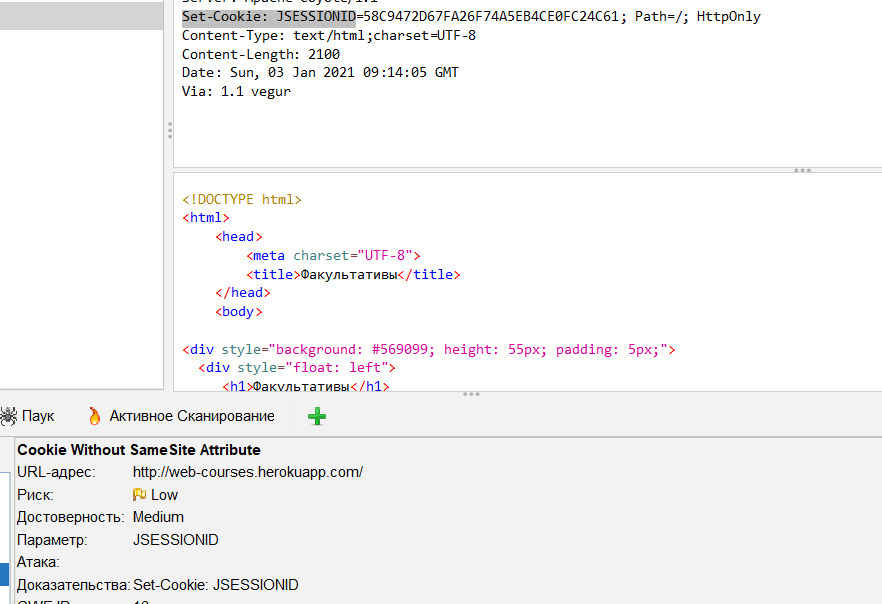
Может быть исправлено путем реализации механизма для предоставления уникальной ссылки / идентификатора ошибки клиенту (браузеру), регистрируя при этом данные на стороне сервера и не раскрывая их пользователю.



1. Cookie Without SameSite Attribute

Файл cookie был установлен без атрибута SameSite, что означает, что файл cookie может быть отправлен в результате межсайтового запроса. Атрибут SameSite является эффективной мерой противодействия подделке межсайтовых запросов, включению межсайтовых скриптов и атакам по времени.

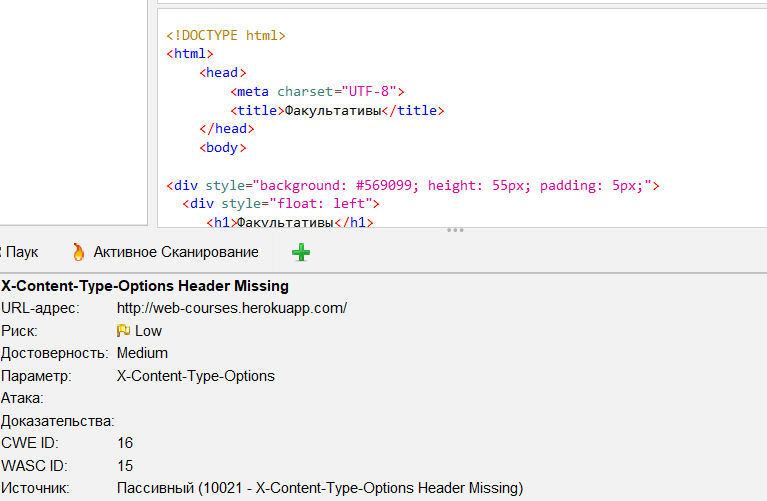
Может быть исправлено путем добавления атрибута SameSite в файл cookie.



1. X-Content-Type-Options Header Missing

Для заголовка Anti-MIME-Sniffing X-Content-Type-Options не задано значение nosniff. Это позволяет более старым версиям Internet Explorer и Chrome выполнять MIME-сниффинг тела ответа, что может привести к интерпретации и отображению тела ответа как тип контента, отличный от объявленного типа контента. Текущая (начало 2014 г.) и устаревшая версии Firefox будут использовать объявленный тип содержимого (если он установлен), а не выполнять сниффинг MIME.

Может быть исправлено путем указания значения nosniff.



После выполнения тестирования были исправлены следующие уязвимости:

X-Frame-Options Header Not Set – обновлена CORS политика;

Absence of Anti-CSRF Tokens – добавлена библиотеки anti-CRSF;

Application Error Disclosure – был добавлен дополнительный обработчик ошибок, который в случае неописанной ошибки возвращает уведомление об ошибке.

Остальные ошибки исправить было сложнее и они несут гораздо меньший риск (Low), поэтому на данном этапе, исходя из принципа «время-деньги», их было решено не исправлять.

