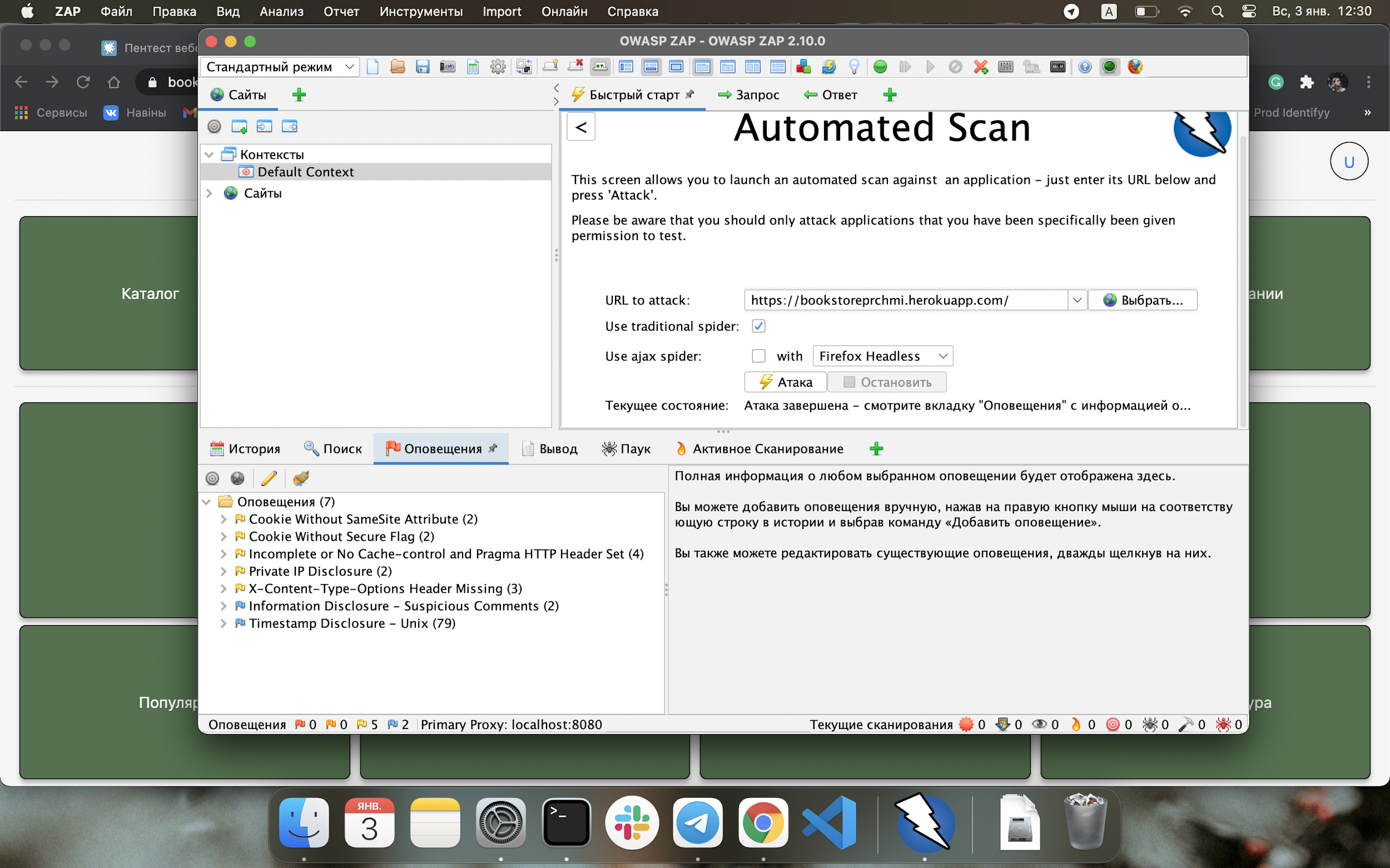
Сайт, который был протестирован: <https://bookstoreprchmi.herokuapp.com/>  
Приложение написано на Ruby on Rails и хостится на Heroku

Результат сканирования:



* Cookie Without SameSite Attribute

Для исправления этой уязвимости воспользуемся встроенным методом Rails - добавим в ApplicationController строку protect\_from\_forgery with: :exception

* Cookie Without Secure Flag

Для исправления необходимо изменить config/initializers/session\_store.rb следующим образом:

MyApp::Application.config.session\_store :cookie\_store, key: '\_my\_app\_session', secure: Rails.env.production?

Это сделает куки приложения более безопасными

* Incomplete or No Cache-control and Pragma HTTP Header Set

Для борьбы с этой уязвимостью Rails предоставляет два метода для определения временного кэширования ресурсов через HTTP-заголовок - expires\_in и expires\_now.

Пропишем для каждой сущности максимальное время жизни в методе show:

expires\_in 3.minutes, :public => true

* X-Content-Type-Options Header Missing

Для исправления этой уязвимости необходимо изменить дефолтный HTTP response header следующим образом:

config.action\_dispatch.default\_headers = {

'X-Frame-Options' => 'SAMEORIGIN',

'X-XSS-Protection' => '1; mode=block',

'X-Content-Type-Options' => 'nosniff'

}

Сканирование после исправления тех уязвимостей, которые получилось исправить:

