УДК 004.62

ВНЕДРЕНИЕ СИСТЕМЫ АВТОРИЗАЦИИ И АУТЕНТИФИКАЦИИ ПОЛЬЗОВАТЕЛЕЙ НА ОСНОВЕ ФРЕЙМВОРКА SPRINGSECURITY В ВЕБ-ПРИЛОЖЕНИИ «ОНЛАЙН СЕРВИСА ДЛЯ ОБУЧЕНИЯ ПРОГРАММИРОВАНИЯ НА JAVA»

П.А. СТАНКЕВИЧ
(Представлено: Ю.Н. КРАВЧЕНКО)

Рассматривается настройка фреймворка SpringSecurity в качестве механизма обеспечения аутентификации и авторизации пользователей для разработанного онлайн сервиса. Данное веб-приложение было реализовано с использованием множества фреймворков, одними из которых стали Spring Framework, Spring Security и MyBatis.

При создании веб-приложений, в которых существуют различные роли пользователей, часто возникает вопрос, каким способом создавать авторизацию, где ее хранить, как аутентифицировать и авторизовывать пользователей. Для этого программисты практически всегда используют различные фреймворки. Об одном из таких фреймворков, а именно SpringSecurity, который был внедрен в онлайн сервис для обучения программированию на Java и пойдет речь.

Основная часть

На стадии проектирования веб-приложения были составлены основные требования для обеспечения аутентификации и авторизации пользователей, а именно:

- взаимодействие с фреймворками SpringFrameworkиApacheCommonsDBCP;
- высокая скорость работы;
- возможность описания файла конфигурации в XML формате.

В процессе анализа возможных вариантов был выбран фреймворк SpringSecurity, так как он полностью соответствует перечисленным требованиям и имеет достаточно детализированные возможности по настройке.

SpringSecurity это Java/JavaEE фреймворк, предоставляющий механизмы построения систем аутентификации и авторизации, а также другие возможности обеспечения безопасности для промышленных приложений, созданных с помощью SpringFramework. Проект был начат Беном Алексом (BenAlex) в конце 2003 года под именем «AcegiSecurity» и был публично представлен под лицензией ApacheLicense в марте 2004. Впоследствии был включен в Spring как официальный дочерний проект. Он был впервые публично представлен под новым именем SpringSecurity 2.0.0 в апреле 2008 года, что включило официальную поддержку и подготовку от SpringSource [1].

«Онлайн сервис для обучения программированию на Java» должен был обладать 3 различными ролями пользователей: модератор, администратор и пользователь.

Пользователь должен был иметь следующие права:

- просмотр и редактирование личной информации;
- просмотр обучающих материалов;
- просмотр магазина и осуществнление покупок за внутри-сервисную валюту;
- прохождение тестов;
- отправление жалоб на вопросы из-за найденных ошибок;
- создание и участие в группах.

Каждому создателю группы было доступно меню просмотра и управления группой.

Модератору должны быть присвоены такие права, как:

- просмотр и редактирование обучающих материалов;
- редактирование цен на внутрисервисную валюту;
- просмотра диаграмм, отражающих актуальность товаров в магазине;
- редактирование количества начисляемой внутри-сервисной валюты за различные действия пользователя;
- просмотр жалоб пользователей;
- подтверждение или отклонение найденной пользователем ошибки.

Администратор должен обладать всеми возможностями модератора, а также возможностью управлять модераторами.

Рассмотрим реализацию конфигурации SpringSecurity в разработанном веб-приложении.

Часть файла конфигурации представлена в листинге 1.

Листинг 1 – Файл конфигурации SpringSecurity

```
37  <?xml version="1.0" encoding="UTF-8"?>
38  <b:beansxmlns="http://www.springframework.org/schema/security"
39  xmlns:b="http://www.springframework.org/schema/beans"
40  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
41  xsi:schemaLocation="
42    http://www.springframework.org/schema/security
43    http://www.springframework.org/schema/security/spring-security.xsd
44    http://www.springframework.org/schema/beans
45    http://www.springframework.org/schema/beans/spring-beans.xsd">
46
47  <http realm="Learn Java" use-expressions="false">
48  <intercept-url pattern="/index.html*" access="IS_AUTHENTICATED_ANONYMOUSLY"
49  />
50  <intercept-url pattern="/login.html*" access="ROLE_ANONYMOUS" />
51  <intercept-url pattern="/logout.html*" access="IS_AUTHENTICATED_ANONYMOUSLY"
52  />
53  <intercept-url pattern="/register.html*" access="ROLE_ANONYMOUS" />
54  <intercept-url pattern="/css/**" access="IS_AUTHENTICATED_ANONYMOUSLY" />
55  <intercept-url pattern="/images/**" access="IS_AUTHENTICATED_ANONYMOUSLY" />
56  <intercept-url pattern="/fonts/**" access="IS_AUTHENTICATED_ANONYMOUSLY" />
57  <intercept-url pattern="/js/**" access="IS_AUTHENTICATED_ANONYMOUSLY" />
58  <intercept-url pattern="/profile.html*" access="ROLE_USER, ROLE_MODER,
59  ROLE_ADMIN" />
...
30  <intercept-url pattern="/addModer.html*" access="ROLE_ADMIN" />
...
38  <intercept-url pattern="/**" access="ROLE_USER" />
39  <form-login login-page="/login.html" username-parameter="email" password-
40  parameter="passwd" authentication-failure-url="/login.html?login_error=auth" />
41  <logout logout-success-url="/logout.html" />
42  <csrf />
43  </http>
44
45  <authentication-manager>
46  <authentication-provider>
47  <password-encoder hash="md5"/>
48  <jdbc-user-service data-source-ref="dataSource"
49        users-by-username-query="
50        SELECT LOWER(username) as username, password, enabled
51        FROM learn_java.login(LOWER(?))"
52        authorities-by-username-query="
53         SELECT LOWER(u.email) AS username, r.name AS role
54         FROM learn_java.users u
55         INNER JOIN learn_java.roles r ON u.id_role=r.id
56         WHERE u.email=LOWER(?)"/>
57  </authentication-provider>
58  </authentication-manager>
59
60  </b:beans>
```

В первой строке файла расположена XML-декларация. XML-декларация не является обязательной. Однако если она существует, то должна располагаться в первой строке документа, и до нее не должно быть больше ничего, в том числе пробелов.

Со второй по девятую строку листинга идет объявление схемы SpringSecurityConfig для XML документа.

В 11 строке начинается описание тега http. В разрабатываемом приложении было использовано всего два параметра, которые являются realm и use-expressions. Атрибут realm указывает, за какую область будет отвечать последующая настройка. Use-expressions включает EL-выражения, по умолчанию они включены.

С 12 по 38 строку листинга показаны примеры для указания ролей к определенным ссылкам. Роли ROLE_USER, ROLE_MODER, ROLE_ADMIN являются внедренными разработчиком, а такие роли как IS_AUTHENTICATED_ANONYMOUSLY, ROLE_ANONYMOUS являются статическими, они реализованы в самом фреймворке. При просмотре документации к SpringSecurity можно прочитать, что IS_AUTHENTICATED_ANONYMOUSLY хранит ссылку, указывающую на ROLE_ANONYMOUS [2].

Тег form-login из 39 строки, используется для указания страницы (атрибут login-page), на которой пользователю будет предоставлена возможность для авторизации, в атрибутах username-parameter и password-parameter указываются значения name полей ввода логина и пароля, а в атрибуте authentication-failure-url можно указать ссылку, куда будет перенаправлен пользователь, если он введет некорректные данные.

В 41 строке указан тег logout, в нем есть возможность указать, каким способом будет происходить выход из системы. Все настройки были оставлены по умолчанию, кроме logout-success-url – это url, на который будет перенаправлен пользователь для выхода из системы. Тег csrf указывает, что необходимо добавить защиту приложения с помощью CrossSiteRequestForger.

Для аутентификации необходимо настроить тег authentication-manager, которому необходимо задать настройки для authentication-provider. Authentication-provider загружает пользовательскую информацию из jdbc-user-service и сравнивает комбинацию имени пользователя и пароля со значениями, указанными при входе в систему, причем можно указать в атрибуте password-encoder, с помощью какого метода хешируется пароль пользователя.

Заключение

Разработка веб-приложений с использованием фреймворка SpringSecurity для осуществления аутентификации и авторизации пользователей в онлайн сервисе является подходящим средством, так как предоставляет надежный, скрытый механизм предоставления данных пользователям, имеет достаточно большое количество всевозможных настроек, большинство из которых не были переопределены, так как настройки имеют по умолчанию имеют жесткий стандарт, подходящий для большинства приложений, а также SpringSecurity полностью взаимодействует с SpringFrameworkчто обеспечило еще большую легкость в его подключении.

ЛИТЕРАТУРА

1. SpringSecurity – Википедия [Электронный ресурс]. – Режим доступа: https://ru.wikipedia.org/wiki/Spring_Security. – Дата доступа: 21.09.2017.
2. SpringSecurityDocs [Электронный ресурс]. – Режим доступа: https://docs.spring.io/spring-security/site/docs/4.2.3.RELEASE/reference/htmlsingle/. – Дата доступа: 21.09.2017.