МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Одеський національний технологічний університет
Університет Інформатики і прикладних знань, м.Лодзь, Польща
Національний технічний університет України «Київський політехнічний інститут»
Навчально-науковий інститут комп'ютерних систем і технологій «Індустрія 4.0» ім. П.М. Платонова

XXII Всеукраїнська науково-технічна конференція молодих вчених, аспірантів та студентів

«СТАН, ДОСЯГНЕННЯ ТА ПЕРСПЕКТИВИ ІНФОРМАЦІЙНИХ СИСТЕМ І ТЕХНОЛОГІЙ»

Матеріали конференції

Одеса

21-22 квітня 2022 р.

Стан, досягнення та перспективи інформаційних систем і технологій / Матеріали XXII Всеукраїнської науково-технічної конференції молодих вчених, аспірантів та студентів. Одеса, 21-22 квітня 2022 р**.** - Одеса, Видавництво ОНТУ, 2022 р. – 251 с.

Збірник включає матеріали доповідей учасників конференції, які об'єднані за тематичними напрямками конференції.

ОРГАНІЗАЦІЙНИЙ КОМІТЕТ

Голова - д.т.н., проф., **Єгоров Б.В.**, ректор ОНТУ

Співголови:

Поварова Н.М. – к.т.н., доц., проректор з наукової роботи ОНТУ,
Котлик С.В. – к.т.н., доц., директор ННІКСіТ "Індустрія 4.0" ОНТУ,
Даріуш Долива, д.математичн.наук, уповноважений декана факультету Інформатики УІтаПЗ, м.Лодзь, Польща,
Ковалюк Т.В. - к.т.н., доц., Київський національний університет імені Тараса Шевченка

Члени оргкомітету:

Плотніков В. М. – д.т.н., проф., завідувач кафедри ІТтаКБ ОНТУ,
Артеменко С.В. – д.т.н., проф., завідувач кафедри КІ ОНТУ,
Хобін В.А. – д.т.н., проф., завідувач кафедри АТПтаРС ОНТУ,
Тарасенко В.П. – д.т.н., проф., завідувач кафедри СКС НТУУ «Київський політехнічний інститут»,
Невлюдов І.Ш. – д.т.н., проф., завідувач кафедри КІТАМ ХНУРЕ,
Мельник А.О. – д.т.н., проф., завідувач кафедри ЕОМ НУ “Львівська політехніка”,
Жуков І.А. – д.т.н., проф., завідувач кафедри КСтаМ НАУ.

Матеріали подано українською та англійською мовами.
Редактор збірника Котлик С.В.

конференція підрозділів Вінницького національного технічного університету (2020)», Вінниця, 2020. [Електронний ресурс]. Режим доступу: https://conferences.vntu.edu.ua/index.php/allvntu/index/pages/view/zbirn2020.

3. William Blair. Database Software Market: The Long-Awaited Shake-up [Електронний ресурс] – Режим доступу до ресурсу: https://blocksandfiles.com/wp-content/uploads/2019/03/Database-Software-Market-White-Paper.pdf

4. Object-relational mapping (ORM) [Електронний ресурс] // TechTarget. – 2008. – Режим доступу до ресурсу: shorturl.at/zEL18.

5. Тестирование производительности Python 3 ORM методом, основанном на бенчмарке TPC-C [Електронний ресурс] // WEN THemes. – 2020. – Режим доступу до ресурсу: https://itworld.uz/blog/testirovanie-proizvoditelnosti-python-orm-metodom-osnovannom-na-benchmarke-tpc-c/.

6. Кондуфоров А. Сравнение производительности .NET ORM: Часть 1. Выборки данных [Електронний ресурс] / Александр Кондуфоров. – 2008. – Режим доступу до ресурсу: 3. http://merle-amber.blogspot.com/2008/11/net-orm-1.html.

УДК 004.622

SPRING SECURITY МОДУЛЬ ЗАХИСТУ JAVA ПРОГРАМ

МАЙДАНЮК В. П., МАРУЩАК А. В.
(maidaniuk2000@gmail.com, maruskhak@gmail.com)
Вінницький національний технічний університет

Ключові слова: Spring 5.0, Spring Security, Java, захист даних, шифрування паролів, автентифікація, авторизація.

Анотація: у даній публікації описано модуль безпеки персональних даних найбільш популярного фреймворку для розробки Java застосунків Spring. На момент написання актуальною є версія 5.6.2. У сучасному світі з розвитком технологій збільшуються випадки втрати особистої інформації, отже важливим аспектом у розробці програмних додатків є забезпечення користувацької безпеки.

Spring Security – це фреймворк, який надає функції безпеки, такі як: автентифікація, авторизація для створення програмних застосунків з використанням мови програмування Java. Авторизація – це процес, що дозволяє розробнику побудувати у запланованому програмному забезпеченні необхідну ієрархію користувачів з різним доступом до виконання дії. Автентифікація – це додатковий процес для успішного проходження авторизації, який повинен забезпечувати правильне розпізнання та ідентифікацію кожного потенційного користувача, який намагається отримати доступ до системи [1].

Фреймворк Spring Security підтримує широкий спектр моделей поведінки, наявна можливість інтеграції з популярними технологіями, такими як: HTTP Basic, LDAP, OpenID, AppFuse.

Перевагами у використанні даного модуля безпеки є повна підтримка автентифікації та авторизації користувачів, опрацювання даних в окремому потоці, інтеграція API Servlet, підтримка Spring MVC, портативність та мультиплатформність, повноцінна підтримка конфігурації Java.

Основний функціонал програмного доповнення Spring Security [1]:

- LDAP (полегшений протокол доступу до каталогів) – це відкритий прикладний протокол для підтримки та доступу до інформаційних служб розподілених каталогів через інтернет-мережу;

- єдиний вхід – ця функція дозволяє користувачеві отримати доступ до кількох програм за допомогою одного облікового запису, отже наявна можливість застосовувати єдиний логін та пароль для доступу до різних ресурсів;

- запам'ятовування користувача – реалізований даний функціонал за допомогою файлів cookie HTTP. Надає можливість системі запам'ятати визначеного користувача та уникати повторного введення персональних даних.

- OAuth 2.0 – ця функція надає користувачеві увійти в програму, використовуючи наявний обліковий запис соціальних мереж, GitHub, Google. Для правильного функціонування даної функції потрібно увімкнути двоступеневу автентифікацію за допомогою коду.

Починаючи з версії Spring Security 5.0, додана можливість забезпечити реактивне програмування та підтримку реактивного веб-виконання, також дана система може інтегруватися з Spring WebFlux.

У поточній версії Spring Security 5.0 було оголошено PasswordEncoder як застарілий [2]. Це був логічний крок, адже такий підхід не був оптимізований для випадково генерованого ключа шифрування. Отже, було змінено спосіб обробки закодованих паролів. У попередніх версіях кожна програма використовувала лише один алгоритм кодування пароля. За замовчуванням виконувалося це за допомогою StandardPasswordEncoder. Для кодування використовувався SHA-256 алгоритм. Для вбудування нового функціоналу використано концепцію делегування кодування пароля. Такий підхід дав можливість використовувати різні кодування для різних паролів. Spring розпізнає алгоритм за ідентифікатором із префікса закодованого пароля. Наприклад, {bcrypt}$2b$12$FaLabMRystU4MLAasNOKb.HUElBAabuQdX59RWHq5X.9Ghm692NEi – пароль закодовано алгоритмом «bcrypt». На початку у фігурних дужках вказано тип використаного алгоритму під час шифрування, за допомогою цього ідентифікатора декодер розуміє як потрібно розшифровувати отриманий код.

Додавання конфігурації делегування паролів у програмне забезпечення є не важким процесом. Якщо хеш пароля не має префікса, процес делегування використовує алгоритм за замовчуванням. Отже, за замовчуванням буде використовуватися StandardPasswordEncoder. Таке рішення робить нові версії програмного забезпечення повністю сумісними із конфігурацією минулих версій. У версії Spring 5 представлено PasswordEncoderFacoryes. createDelegatingPasswordEncoder(). Даний вбудований метод повертає налаштований екземпляр класу DelegationPasswordEncoder [2]. Для паролів без префікса буде виконуватися поведінка за замовчуванням, а для хешів паролів, які містять префікс, делегування виконується відповідно передбаченого алгоритму. У Spring Security 5.0 додано такі методи шифрування [2]: bcrypt – BcryptPasswordEncoder; ldap – LdapShaPasswordEncoder; MD4 - Md4PasswordEncoder; MD5 - new MessageDigestPasswordEncoder("MD5"); noop – NoOpPasswordEncoder; pbkdf2 - Pbkdf2PasswordEncoder; scrypt – ScryptPasswordEncoder; SHA-1 - new MessageDigestPasswordEncoder("SHA-1"); SHA-256 - new MessageDigestPasswordEncoder("SHA-256"); sha256 – StandardPasswordEncoder; argon2 - Argon2PasswordEncoder. Звичайно, було передбачено, що алгоритм виконання можна змінювати. Наприклад, є задача, де:

- bcrypt – нове значення за замовчуванням;
- scrypt – альтернатичний алгоритм;
- SHA-256 – поточний алгоритм.

Для такого випадку конфігураційний метод програмного застосунку буде мати структуру, як показано на рисунку 1.

```
@Bean
public PasswordEncoder delegatingPasswordEncoder() {
    PasswordEncoder defaultEncoder = new StandardPasswordEncoder();
    Map<String, PasswordEncoder> encoders = new HashMap<>();
    encoders.put("bcrypt", new BCryptPasswordEncoder());
    encoders.put("scrypt", new SCryptPasswordEncoder());

    DelegatingPasswordEncoder passworEncoder = new DelegatingPasswordEncoder(
      "bcrypt", encoders);
    passworEncoder.setDefaultPasswordEncoderForMatches(defaultEncoder);

    return passworEncoder;
}
```

Рисунок 1 – Конфігурація делегування паролів

Отже, розглянуто потужний фреймворк для побудови застосунків з використання мови програмування Java. Детально проаналізовано переваги та використовувані технології захисту інформації, персональних даних з використанням модуля Spring Security. Розглянуто новий функціонал кодування паролів, який є доступний у поточній версії Spring 5.6.2. Внесено зміни у стандартну конфігурацію програмного модуля Spring Security та отримано індивідуальний алгоритм обробки паролів, який надав змогу обробляти вхідні хеші паролів з урахуванням конфігурації програмного забезпечення.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

[1] Spring Security. [Електронний ресурс]. – 2022 – Режим доступу до ресурсу: https://docs.spring.io/spring-security/reference/index.html

[2] What is Spring security. [Електронний ресурс]. – 2021 – Режим доступу до ресурсу: https://www.javadevjournal.com/spring/what-is-spring-security/

УДК 004.056.5

УПРАВЛІННЯ ЗАХИСТОМ ІНФОРМАЦІЙНО-АНАЛІТИЧНОЇ СИСТЕМИ (ІАС) ПРИЙМАЛЬНОЇ КОМІСІЇ ОНТУ (ОНАХТ)

МОРОЗ А.М., ПОХЛЄБІНА Н.О.
ОНТУ (Україна)

Анотація

В роботі розглянуто особливості захисту інформаційних ресурсів приймальної комісії, яка використовується, як засіб обробки та зберігання персональних даних абітурієнтів та вже студентів, що вже вступили до університету, що значно полегшує роботу приймальної комісії та прискорює якість надання даних до відділу навчання. Було виявлено основні недоліки та розглянути шляхи для удосконалення системи безпеки та збереження повної конфіденційності персональних даних вступників. Представлена схематична структура часткового алгоритму із захисту системи.

Ключеві слова: персональні данні, захист інформаційних баз, інформаційно-аналітична система

Проблема захисту є багатоплановою і комплексною і охоплює низку важливих завдань. Проблеми інформаційної безпеки постійно посилюються процесами проникнення в усі сфери суспільства технічних засобів обробки та передачі даних та, насамперед, обчислювальних

XXII Всеукраїнська науково-технічна конференція молодих вчених, аспірантів та студентів

«СТАН, ДОСЯГНЕННЯ ТА ПЕРСПЕКТИВИ ІНФОРМАЦІЙНИХ СИСТЕМ І ТЕХНОЛОГІЙ»

Одеса

21-22 квітня 2022 р

Збірник включає доповіді учасників конференції. Тези доповідей публікуються у вигляді, в якому вони були подані авторами.

Відповідальність за зміст і форму подачі матеріалу несуть автори статей.

Редакційна колегія: Котлик С.В., Корнієнко Ю.К.

Комп'ютерний набір і верстка: Соколова О.П.

Відповідальний за випуск: Котлик С.В.