УДК 65.012

Гетьман В.В.

ПРОЄКТУВАННЯ ТА РОЗРОБКА СЕРВІСНОЇ АРХІТЕКТУРИ УПРАВЛІННЯ БІЗНЕС-ПРОЦЕСАМИ УНІВЕРСИТЕТУ. БЕЗПЕКА СИСТЕМИ

Державна політика в Україні в галузі освіти постійно реформується і розвивається. Одним із шляхів цього розвитку є модернізація інфраструктури інформаційного освітнього середовища. Для реалізації цього напрямку постає необхідність створення спеціалізованих електронних ресурсів ЗВО. Проведений аналіз відкритих електронних ресурсів України і світу, демонструє, що у більшості ЗВО використовуються та періодично доопрацьовуються інформаційні ресурси; існують окремі малоінформативні сайти або з повною відсутністю функціоналу для певних груп користувачів (викладачів, студентів, персоналу, тощо). Отже, нагальною є потреба розробки на рівні держави типової структури побудови веб-сайту ЗВО з визначенням обов'язкових елементів та дотримання найсучасніших підходів до розробки.

Ключові слова: управління, інформаційна безпека, розробка.

One of the directions of modern state policy in Ukraine in the field of education is to improve the infrastructure of information educational space. In this context, it is important to create appropriate electronic resources for the university. The analysis of open electronic resources of Ukraine and the world allows us to draw the following conclusions: in the vast majority of universities, information sites have been created and periodically improved; there are some sites with a low level of information or inconsistency in the needs of certain categories of users (employees, students, entrants, etc.). Therefore, there is an urgent need to develop at the state level a standard structure for building a university website with the definition of mandatory elements and adherence to the latest approaches to development.

Keywords: management, information security, development.

Після проведення порівняльного аналізу систем управління діяльності ЗВО, які виявилися найбільш вживаними університетами світу , було виявлено, що нинішній стан використання інформаційних технологій в управлінні діяльністю ЗВО має ряд нагальни проблеми. Результати аналізу, узагальнення та систематизації напрацювань ЗВО та IT компаній спрямованих на модернізацію наукової, навчальної та організаційної діяльності ЗВО, визначення вимог до інформаційних систем управління процесами ЗВО мають велику інформаційну цінність. Базуючись на цих данніх, постає потреба в створенні та правильному використання засобів та інструментів для забезпечення безпеки та цілісності цих систем. Для реалізації цих вимог, було розроблено наступний алгоритм попереднього аналізу ЗВО:

а) Проаналізувати характеристики існуючих систем, зокрема обсяг їх можливостей.

б) Проаналізувати окремі бізнес процеси.

в) На основі проведеного аналізу розробити вимоги щодо можливостей серверної частини системи.

г) Відповідно до створених вимог розробити серверну частину, зокрема реалізувати структуру бази даних та публічний API.

ґ) Спланувати та розробити систему заходів безпеки для даної системи

д) Розробити документацію до публічного API.

е) Обґрунтувати використані технології при проектуванні серверної частини. Очікується, що спроектований продукт буде придатний до використання всіма учасниками освітнього процесу в закладах вищої освіти.

З точки зору програмного забезпечення, система містить в собі певний масив елементів, які контролюють доступ користувачів до файлів, розміщених на сервері, наприклад - HTTP-сервер. Сервери кешують веб-сторінки та надають їх користувачу за запитом, надісланим через HTTP, якій у всій структурі відіграє ключову роль як протокол, відповідальний за передачу данних від клієнта до сервера, і навпаки.

Очевидно, що мережа вразлива до атак з боку зловмисників, які ведуть свою діяльність використовуючи безліч способів. Тому будь-яка прогалина в системі захисту програми, бази даних, операційної системи або в самій мережі може привести до атаки на веб-сервер. Атакою на WEB-сервер – це порушення звичної роботи компонента, видалення або модифікація його даних або отримання невідповідного доступу до операційної системи.

Вхід користувача в систему є чи не найбільш вразливим моментом у системі. Даний етап вміщує у себе основні функції передачі та прийому даних від інших сервісів через інтерфейси API та передачу JSON даних на клієнтській стороні з використанням Javascript фреймворка ReactJS. Основними кроками авторизації користувача в системі є:

1. Вхід користувача на веб-сайт сервісу;
2. Введення користувацького пароля та елекронної пошти в відповідній формі входу;
3. Вибір ролі користувача, яка буде надавати певні привілеї або обмеження як на функціонал так і на отримувані дані.

При вході у профіль користувач може поміняти свої дані, такі як пароль, опис профілю та інше. При цьому відбувається передача данних до сервісу авторизації та аутентифікації. Оскільки для даного додатку необхідно зберігати та отримувати дані, то було вирішено додати сервіс авторизації та аутентифікації, що зберігає дані про користувача, а також видає ключі для використання інших сервісів у проекті. Для процесу авторизації найбільш якісним є JSON Web Tokens (JWT). JWT не є новим механізмом, проте дуже гарно підходить для мікросервісної архітектури. Для того, щоб використовувати сервіси прогнозування та аналітики користувачеві необхідно в відповідні формі ввести свій пароль та електронну пошту, в результаті сервіс користувацького інтерфейсу відправить відповідний POST запит на генерацію JWT точену для користувача. З даним токеном користувач є авторизованим в системі, що дозволяє йому використовувати всі можливості системи. На рис. 1 зображено структуру типового JWT токена:

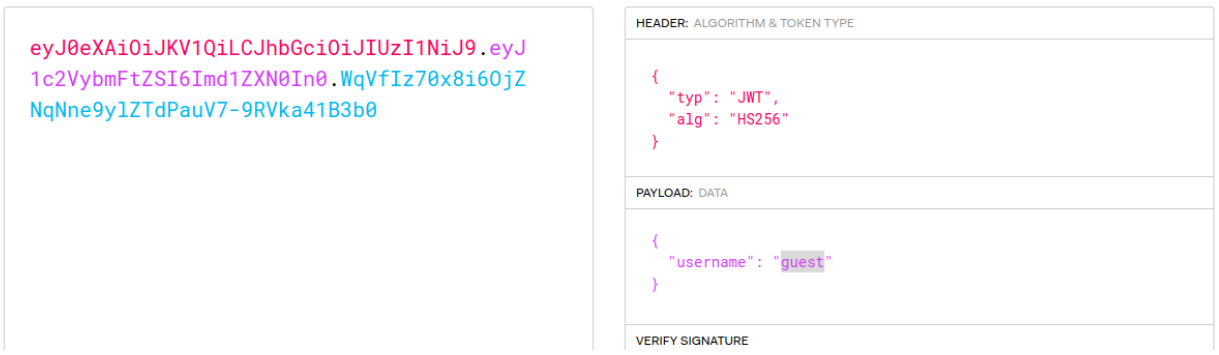

Рис. 1. Структура JWT токена

Кожен токен складається з трьох компонентів розділених крапкою, кожен з яких має свої функції при розпізнанні сервером. Червоним кольором на малюнку позначено хедер токену, в якому збережена інформація про алгоритм, що було використано для шифрування секретної частини токену, хедер – не що інше як JSON об’єкт, а сам хедер надалі форматується у base64 строку як показано на рисунку. Наступним елементом є тіло токену в якій і зберігаються дані користувача, виділена фіолетовим кольором на рисунку. В даній частині можуть зберігатися ролі користувача у системі, пошта, ім’я і т.д. Ця частина токену має вигляд об’єкта JSON і форматується як base64, тобто її можно

розшифрувати на клієнті у разі необхідності і отримати необхідні дані про користувача. Найважливіша частина токену виділена синім, ця частина є гарантією того, що серверу передали достовірний токен і саме з цією частиною працює сервер авторизації для перевірки даних доступу користувача до компонентів системи. Вона формується з використанням алгоритму, зазначеного в хедері і розраховується так:

$$T = Algo(base64(header) + "." + base64(signature), secretKey)$$

Виходячи з формули, сервер повинен мати секретний ключ для генерації цієї частини токену, з допомогою якого він зможе розшифрувати секретну частину за алгоритмом у хедері та перевірити правильність отриманих частин токену. Однак, це не єдина можливість даного сервісу, оскільки він відповідає не лише за генерацію і перевірку токену користувача, а й збереження та оновлення даних користувача. Користувач має можливість змінювати пароль у системі, інші дані, а також видаляти свій акаунт із системи, оновлювати основну інформацію про себе, а також переглядати профілі інших користувачів, якщо це необхідно, звичайно, якщо у користувача є відповідні права, для цього на даному сервісі реалізовано такі ендпоінти для використання іншими компонентами. Основні ендпоінти сервісу авторизації та аутентифікації зображені на рис.2.

Метод	URL	Дані	Відповідь
POST	/auth/signin	Пароль та електронна пошта	Згенерований токен
GET	/auth/verify	Токен користувача	Результат перевірки
GET	/user?id	Id користувача	Дані про користувача
POST	/user/	Інформація про користувача	Результат реєстрації
PUT	/user/	Оновлена інформація	Результат оновлення
DELETE	/user/	Id користувача	Результат видалення

Рис. 2. Ендпоінти сервісу авторизації та аутентифікації

Важливим етапом створення безпечної системи було делегування повноважень різним типам користувачів. Спираючись на мікросервісну архітектуру системи було створено декілька окремих ролей, які мали б окремий функціонал та певні обмеження при отриманні даних. Детальніше ці ролі можна роз'яснити пройшовши по основним структурним одиницям університету.

Беручи до уваги величезну кількість технологій, що є компонентами веб – сервера, проведення аналізу можливих атак на веб-сервер та методів їх запобігання є нагальною потребою. З появою нових технологій, захист та атаки на системи регулярно модифікуються та покращуються, тому аналіз деталей реалізації та можливих проблем потребує подальших досліджень.

ЛІТЕРАТУРА:

1. Співаковський О. В. Web-портал Херсонського державного уні-верситету. — 2013. — URL: http://kspu.edu/(дата зверн.11.04.2019).
2. Sankar R. Burpsuite – A Beginner's Guide For Web Application Security or Penetration Testing / Ravi Sankar. – 2018. – URL: https://kalilinuxtutorials.com/burpsuite/.
3. Ricca F. https://dl.acm.org/citation.cfm?id=381476 [Електронний ресурс] / F. Ricca, P. Tonella. – 2001. – URL: https://dl.acm.org/citation.cfm?id=381476
4. Ganore P. What Is A Web Server And How Does It Function? / Pravin Ganore. – 2017. – URL: https://www.milesweb.com/blog/hosting/web-server-function/.

5. How a Web server functions? – 2006. – URL: https://www.eukhost.com/blog/webhosting/how-a-web-serverfunctions/

Рекомендує до друку науковий керівник професор Співаковський О.В.