UDC 004.056

Dmytro Pylypiuk,
Oleksii Aleschenko

AUTHENTICATION METHODS IN WEB APPLICATIONS

Дмитро Пилип'юк,
Олексій Алещенко

СПОСОБИ АВТЕНТИФІКАЦІЇ КОРИСТУВАЧА У WEB-ДОДАТКАХ

The article is considering different methods of user authentication in web applications.

Key words: authentication, web application.
Fig.:5. Tabl. 0. Bibl. 4

У статті розглядається, як web-додатки проводять автентифікацію користувачів, використовуючи різні методи.

Ключові слова: автентифікація, web-додаток.
Рис.:5. Табл. 0. Бібл.:4.

The relevance of the research topic.

The topic of user authentication is widespread in the sphere of web application development and has become increasingly relevant in recent years, since the volume of private data transmitted over the Internet increases over time.

Target setting. Nowadays we use a certain amount of different applications and web resources, that make life easier. However, in order to receive a range of services, we provide private information about ourselves: phone number, address, email, bank card numbers, etc. Each user would prefer that such information be kept confidential and nobody other than the user could access it. This problem has begun development of various mechanisms of authorization and authentication of users. Let's consider how modern services conduct authentication of users and leave access to private data only to the user himself.

Actual scientific researches and issues analysis.

The source [1] addresses the issues of user authentication and describes the authentication protocols that are common at this time.

In article [2], a method is proposed for authenticating users of computer systems, resistant to spying attack, based on a graphic password and a gesture (move) selected by the user, similar to the movements of chess pieces.

The opportunity of two-factor authentication usage in the control systems and access management on the basis of Quick Response codes with one-time passwords is analyzed in the work [3].

Uninvestigated parts of general matters defining.

There is no system that uses an authentication method that is completely safe. Each authorization system can be attacked and allow hackers to steal data.

The research objective.

The purpose of this article is to analyze the most popular methods of user authentication.

The statement of basic materials.

1. Password authentication.

This method consists in the fact that the user must provide the system with a pair of login / password that was specified during registration for successful identification / authentication. This pair is specified when creating a user account on the system. There are standard password authentication protocols that can be applied in web applications.

1.a. HTTP authentication.

This protocol is described in HTTP 1.0 / 1.1 and is applicable in the corporate sphere. The principle of work is as follows:

1. When accessing an unauthorized user, the server returns the "401 Unauthorized" HTTP status and adds the "WWW-Authenticate" header with the specified parameters and the authentication scheme.

2. When receiving such an answer from the server, the browser automatically displays the form of entering the necessary parameters that the user can enter in order to access the resource.

3. In all subsequent user requests for this web resource, the browser automatically adds the HTTP header "Authorization", which transfers the data specified by the client when authorizing.

4. The server authenticates the user according to the data from this header.

Http authentication has several different schemas that differ in security:

1. Basic - the simplest, the parameters are transmitted in the header in unencrypted form. Relatively safe when using HTTPS.

2. Digest - is a schema where a server sends a unique "nonce" value, and the browser sends the MD5 hash to a user's password that was calculated using the "nonce" value. A more secure schema than Basic, but may be striked by the "man-in-the middle" attacks. Also, this scheme is not designed to use modern hash functions to store passwords on the server.

3. NTLM or Windows authentication - as well as Digest, based on the challenge-response principle, in which the user password is transmitted in encrypted form. Not an HTTP standard, but is supported by most browsers and servers. It is mainly used to authenticate Windows Active Directory users in web applications. Sensitive to "pass-the-hash" attacks.

It's worth noting that when using HTTP authentication, the user does not have the standard ability to exit the web application, except to close all browser windows.

2.Certificate authentication.

The certificate is a set of attributes that identifies the user and is signed by the certificate authority (CA). CA acts as an intermediary, which guarantees the authenticity of certificates. Also, the certificate is cryptographically associated with a private key, which is stored by the certificate owner and confirms the fact of possession of the certificate.

On the client side, the certificate may be stored along with the private key in the operating system, in the browser, in the file on the physical device. The private key is additionally protected by the password.

Web applications traditionally use certificates X.509. Authentication with such certificates occurs at the time of connection to the server and is part of the SSL / TLS protocol.

During authentication, the server performs validation based on the following rules:

1. Certificate signed by CA.
2. The certificate has not expired.
3. The certificate shall not be withdrawn by the relevant CA.

After a successful authentication, the web application can execute an authorized request based on the certificate parameters.

Using certificates is a much more reliable way than password authentication. In the process of authentication, a digital signature is created, the presence of which proves the fact of using the private key. However, problems with the distribution and support of certificates make this method of authentication unpredictable in the sphere of information technology.

3.Token authentication.

Tokens are created by the server, signed by a secret key and passed to the client, who in the future uses a token for authentication. There are several standards for web tokens. Consider the most common - JWT - JSON Web Token - a standard for creating access tokens based on the JSON format.

The JWT Token consists of three parts:

1. Header - specifies the information needed to describe the token itself (encryption algorithm, token type, type of content).
2. Payload - is a set of fields where user information (name, level of access, role) is specified.
3. Signature - is generated using encryption algorithms and is calculated based on the first two token blocks.

Tokens are divided into 2 types and perform important roles in the authentication of users of the client-server application:

– Access token is a token that gives its owner access to secure server resources. Usually, it has a short life time and contains additional information.

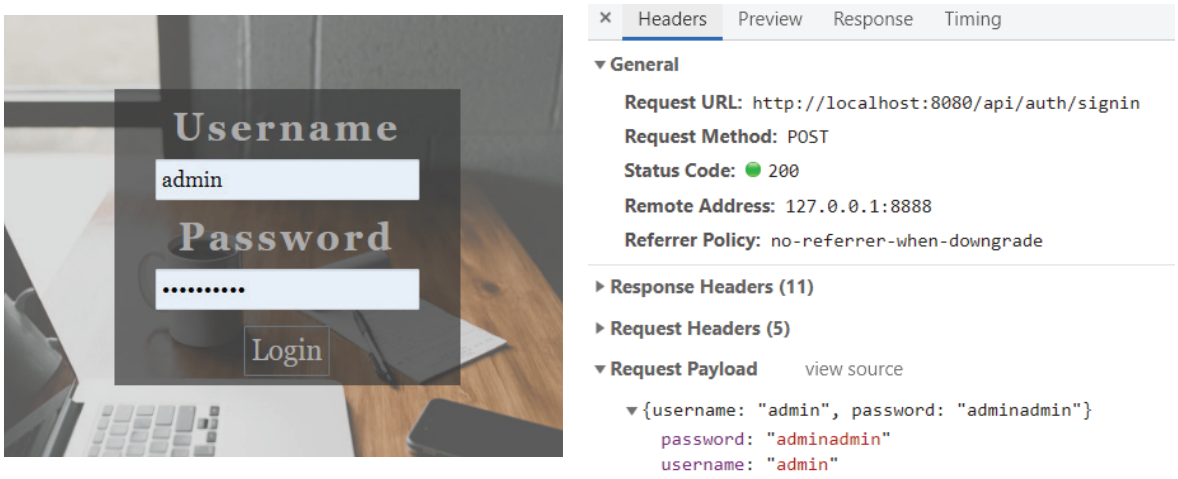
– Refresh token - this token allows clients to request new access tokens after their lifetime ends. These tokens are issued for a long time.

Using tokens in client-server applications:

1. The client is initially authenticated.
2. If the authentication was successful, then the server sends access and refresh tokens to the client.
3. During subsequent queries to the server, the client uses the access token. The server validates validity and provides access to resources.
4. If the access token is not valid, then the client sends a refresh token, in response the server will update both tokens.
5. If the refresh token is not valid then the client must pass the initial authentication process again.

Experiments. It was created its own authorization and authentication system based on access tokens. The tokens, among other things, contain information about the role of the user, which affects the reaction of the system when interacting with it. Tokens are also used as a password encryption mechanism. They contain a hash for the user password, which provides an alternative way to save it in a secure form.

Login frame of the created system and authentication query details are shown in the figure 1. There are user name and password in request payload. There is token and other technical information in the response (see fig. 2). After login user can see landing page in the figure 3.



A B

Fig.1. A – login frame, B – headers of authentication query

Conclusions. This article describes methods of user authenticate in web applications that can be considered fundamental. Among all of these mechanisms, attention was focused on token authentication. This method is more reliable than all of the above. Access Tokens is a much safer mechanism than HTTP authentication, since JWT is almost non-attackable and allows you to store encrypted data in a database. Also, tokens are more practical than certificates, because they exist only during the application works and they need not be maintained as certificates. Tokens are also the only authentication mechanism that allows you to build an SSO (Single Sign-On)

system, where one application allows you to switch to another without re-authentication (like Gmail and YouTube).

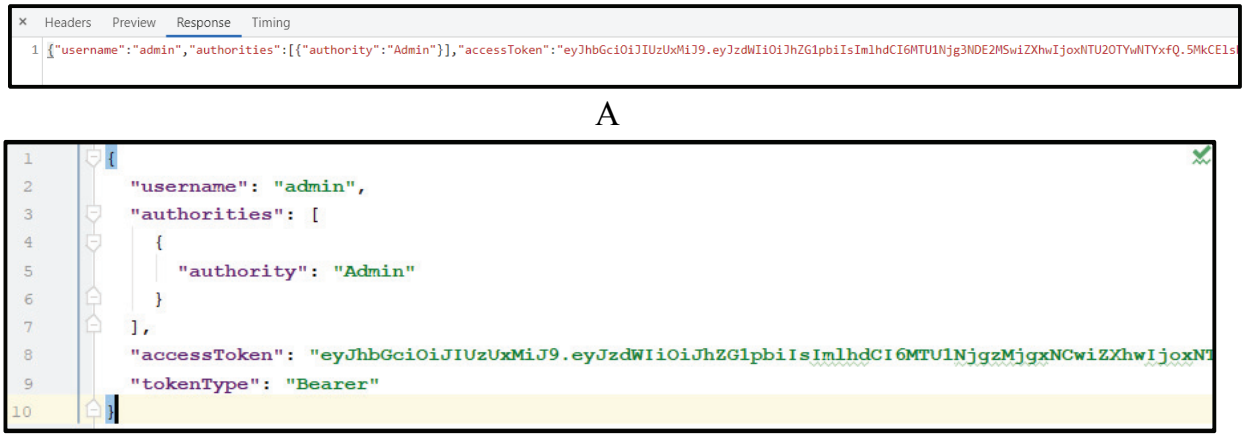


Fig.2. A – authentication response in browser tool view,
B – formatted authentication response

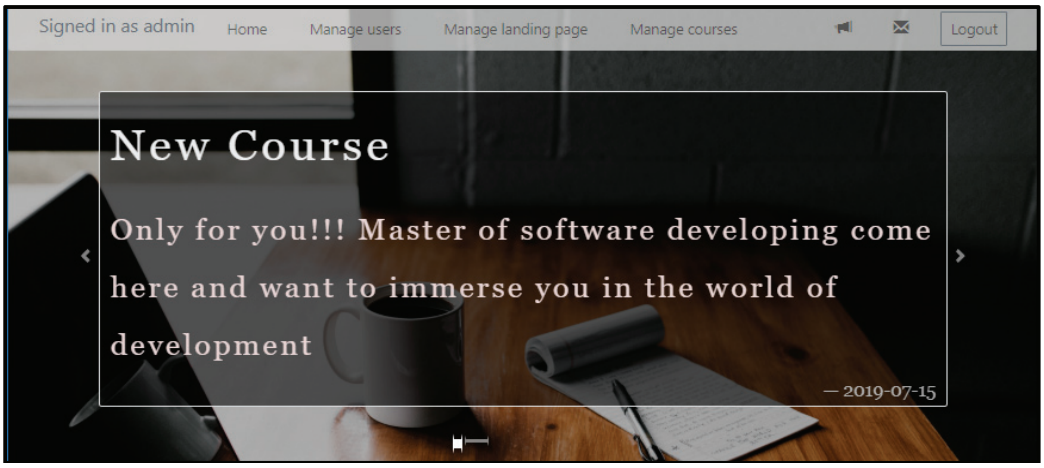


Fig.3. Landing page of the system after login

References

1. Молдовян А. А., Молдовян Д. Н., Левина А. Б. (2016). *Протоколы аутентификации с нулевым разглашением секрета.* (pp. 3-29).

2. Яковлев В. А., Архипов В. В., (2014). *Аутентификация пользователей на основе устойчивого к подсматриваниям графического пароля «Шахматы».* (pp. 25-35).

3. A. Y. Iskhakov (2013). *Two-Factor Authentication System based on QR-Codes.* (pp. 97 – 101).

4. *«Обзор способов и протоколов аутентификации в веб-приложениях»* [Electronic source] (2015) – Access mode: https://habr.com/ru/company/ /dataart/blog/262817/.

Authors

Pylypiuk Dmytro – bachelor student, Department of Computer Engineering, National Technical University of Ukraine “Igor Sikorsky Kyiv Polytechnic Institute”.
E-mail: pylypyuk.dmytro@gmail.com

Пилип’юк Дмитро Олександрович – студент ІІІ курсу, кафедра обчислювальної техніки, Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського».

Aleshchenko Oleksii – senior lecturer, Department of Computer Engineering, National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute".
E-mail: alexey.aleshchenko@gmai.com
Алещенко Олексій Вадимович – старший викладач, кафедра обчислювальної техніки, Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського».

РОЗШИРЕНА АНОТАЦІЯ

Д. О. Пилип’юк,
О. В. Алещенко

СПОСОБИ АВТЕНТИФІКАЦІЇ КОРИСТУВАЧА У WEB-ДОДАТКАХ

Актуальність теми дослідження. Тема автентифікації користувачів є поширеною в сфері розробки web-додатків і в останні роки стає все більш актуальною, оскільки об`єм приватних даних, що передаються через Інтернет, збільшуються з часом.

Постановка проблеми. В сучасному світі ми користуємось певною кількістю застосунків та онлайн ресурсів, які спрощують наше повсякденне життя. Однак для того, щоб отримувати певний спектр послуг, ми надаємо приватну інформацію про себе: номер телефону, адресу, електронну пошту, номери банківських карт тощо. Кожен користувач волів би, щоб така інформація залишалась конфіденційною і ніхто крім самого користувача не міг мати доступ до неї. Дана проблема дала початок розвитку різних механізмів авторизації та автентифікації користувачів. Розглянемо, як сучасні сервіси проводять автентифікацію користувачів і залишають доступ до приватних даних лише самому користувачу.

Аналіз останніх досліджень і публікацій. Протягом останніх років з'являється все більше статей, що зосереджують увагу на нових протоколах і механізмах автентифікації користувача. Окрім автентифікації в Інтернеті, стає популярною тема біометричної автентифікації.

Виділення недосліджених частин загальної проблеми. Немає ні одної системи автентифікації, що була би повністю безпечною. Кожна система може бути вражена атаками і це дозволить хакерам викрасти дані.

Постановка завдання. Метою даної статті є аналіз найбільш популярних методів автентифікації користувачів.

Викладення основного матеріалу. Проведено аналіз трьох методів автентифікації користувачів. Розглянуто принцип їх роботи, переваги використання та недоліки, слабкі місця. Наведений приклад застосування авторизації за допомогою токенів на готовому програмному продукті.

Висновки. В даній статті було розглянуто способи автентифікації користувачів у web-додатках, які можна вважати фундаментальними. Серед усіх зазначених механізмів, увагу було зосереджено на автентифікації через токени. Даний спосіб є надійнішим, ніж усі вище зазначені. Токени доступу є набагато безпечнішим механізмом, ніж HTTP-автентифікація, оскільки JWT майже не підлягає атакам і дозволяє зберігати зашифровані дані у базі даних. Також токени більш практичні з сертифікати, оскільки вони існують лише під час роботи додатку і їх не потрібно підтримувати, як сертифікати. Також токени це єдиний механізм автентифікації, який дозволяє побудувати SSO (Single Sign-On) систему, де один додаток дозволяє перейти в інший без повторної автентифікації (наприклад Gmail і YouTube).

Ключові слова: автентифікація, web-додаток.