PASSWORDLESS AUTHENTICATION USING MAGIC LINK TECHNOLOGY

I.S. Matiushin, V.V. Korkhov[a]

Saint Petersburg State University, 7-9 Universitetskaya emb., Saint Petersburg, 199034, Russia

E-mail: [a] v.korkhov@spbu.ru

Nowadays, the problem of identification and authentication on the Internet is more urgent than ever. There are several reasons for this: on the one hand, there are many Internet services that keep records of users and differentiate their access rights to certain resources; on the other hand, cybercriminals' attacks on web services have become much more frequent lately. At the same time, in many cases, the weak point of systems exposed to attacks is precisely the authentication system.

Authentication methods based on the knowledge factor (e. g. password protection) are the most common and are applied almost everywhere. Their advantages are ease and low cost of implementation. On the other hand, such systems are often vulnerable to various kinds of attacks. It is estimated that up to 80% of successful hacker attacks (including attacks on the largest services with millions of users) succeeded precisely because of the weakness of the password protection system.

This paper presents a solution to the problem of passwordless authentication, which can be applied in a number of online services and systems. In particular, we consider the magic link technology and present an authentication system implemented using Keycloak, an open-source software product that implements single sign-on technology. In the future, it is possible to further improve the system, in particular, using adaptive authentication, which allows switching between different authentication mechanisms depending on certain factors.

Keywords: authentication, passwordless, magic link technology

Iurii Matiushin, Vladimir Korkhov

1. Introduction

Modern distributed systems are becoming more and more complicated, with the number of their users constantly increasing; at the same time, attacks on such systems have become much more frequent as of late. This means that there is a need for security systems (including user authentication systems) that are, on the one hand, reliable and simple to use, and, on the other hand, meet the high security requirements necessary to protect the users' data from cyber threats [1]. Another important aspect is making distributed systems convenient to use; this includes authentication systems.

In general, to access any protected resource, a user needs to go through three consecutive stages:

- Identification – the user must provide an identifier of some kind (username, e-mail address, phone number, etc.)
- Authentication – the user must prove their identity, i.e., provide some form of proof that they are who they claim to be
- Authorization – if authentication is successful, the system grants the user access to the resource

Our work is concerned with user authentication, as the most security-critical part of the entire access-granting pipeline.

Over the years, multiple various authentication methods have been created. They are generally classified according to the factor used to confirm the user's identity. In particular, the three main factors which are recognized nowadays are as follows:

- Knowledge factor ("what you know"), when the user is assumed to possess a piece of secret information, such as a password or a PIN code
- Possession factor ("what you have"), when the user must possess a unique physical device, such as a mobile phone or an electronic key
- Inherence factor ("who you are") refers to methods based on biometrics, such as fingerprint scans, face recognition technology, etc.

Today, knowledge-based authentication – in particular, password protection – is by far the most common authentication method. It has a number of advantages, such as being familiar to most if not all users and being cheap and easy to implement. That said, password-based systems also have several serious disadvantages. For one, there is an entire class of attacks based on obtaining or brute-forcing passwords, and it is estimated that up to 80% of successful hackings succeeded precisely because of a weakness in a password-based security system [2]. Another problem is that password protection can be inconvenient for the end users. For example, based on the survey conducted by Keeper Security, 76% of mobile users store passwords by remembering them or writing them down, which often leads to passwords being forgotten or lost. As a result, 33% of users have to take 3 to 4 login attempts to remember a password, and 60% of users have had to reset a password in the 60 days preceding their participation in the survey [3].

In this paper, we consider passwordless authentication methods. Systems based on such methods have a number of advantages – ease of use, protection against many common types of attacks, and the lack of need to create a large number of passwords. Passwordless authentication technologies are increasingly widespread, and are already in use by a number of large companies – Google, Medium, etc. In particular, the magic link technology is considered. Using it, the end user does not need to use a password to register or log in to the system – just to enter an email address and follow the link sent by the authentication system. The link is unique, and authorization with its help is possible only for a specific user and only for a limited time. This approach not only greatly simplifies the process of registering new users and relieves them of the need to remember passwords, but also provides reliable protection against a number of attacks related to password theft or brute-force attacks.

We believe that the "traditional" password protection systems are often inadequate to the task of secure user authentication in distributed systems. For that reason, we have looked into several alternative authentication methods, and have implemented two such methods in practice.

2. Alternative authentication methods: MFA, 2FA, passwordless authentication

One alternative that we have investigated is the use of multi-factor authentication (MFA).

As the name suggests, MFA means using several authentication methods, based on various authentication factors, to confirm a user's identity. The most common implementations of MFA tend to take the form of two-factor authentication (2FA), making user authentication a two-step process.

One possible implementation of the 2FA concept includes the use of one-time passwords (OTPs). In 2FA systems that use OTPs, the first step of the authentication process is usually similar to a "traditional" password-based authentication: a user enters a username and a password. During the second step, however, the user has to enter an OTP from their mobile device: the OTP is either sent to user's phone number as a text message or generated on the user's phone by a special application. Thus, such a system combines knowledge-based authentication (using passwords) with possession-based authentication (a user having a mobile device which provides the OTPs).

2FA systems possess the advantage of having increased security compared to the "traditional" password-based authentication. For that reason, they are commonly used in contexts where security is crucial (for instance, bank transactions). On the other hand, such systems do not increase the user convenience of the authentication process; on the contrary, having to constantly use a mobile phone to log in can be tiring on everyday basis.

Another option is to forgo passwords entirely, and switch to passwordless authentication.

There are currently several ways to authenticate users without having to use passwords. One option, which has been gaining popularity lately, is to use the magic link technology.

In a system that uses that technology, a user only needs to enter an e-mail address to log in or sign up. An e-mail containing a unique link (which is referred to as "magic link") is then sent to the provided address, and the user can authenticate by simply clicking on the link. The magic link is, in a way, similar to an OTP, since it can be used to authenticate only a certain user and only for a short amount of time.

The magic link technology is easy and convenient for the users. There is no more need for them to remember a large number of passwords; furthermore, since multiple types of cyberattacks are based on obtaining passwords (as established earlier), a system using the technology is protected against many common attacks.

The magic link technology is already used by a number of companies, including Google, Medium, and others.

3. Practical implementations

Currently, we have implemented a passwordless authentication system based on the magic link technology, as well as a 2FA system which uses time-based one-time passwords (TOTPs).

In order to implement these systems, we have used Keycloak. Keycloak is an open-source single sign-on (SSO) software product; SSO means that it allows a user to log in once and access several related services, instead of having to log in multiple times.

We have created a Keycloak magic link authentication module. The module allows the Keycloak administrator to set up e-mail-based passwordless authentication for a certain user, or a group of users.

The authentication flow can be described as follows [4]:

- The user starts the authentication process
- The system requests the user's e-mail address, which the user provides
- If the user doesn't exist within the system's database, a new user is created
- The system then generates a unique token for the magic link and forms the magic link
- The magic link URL is sent to the user's e-mail
- The user clicks on the magic link and is redirected to the authentication page
- The system checks if the magic link is valid; if it is, the user successfully logs in

The authentication flow is shown in Figure 1.

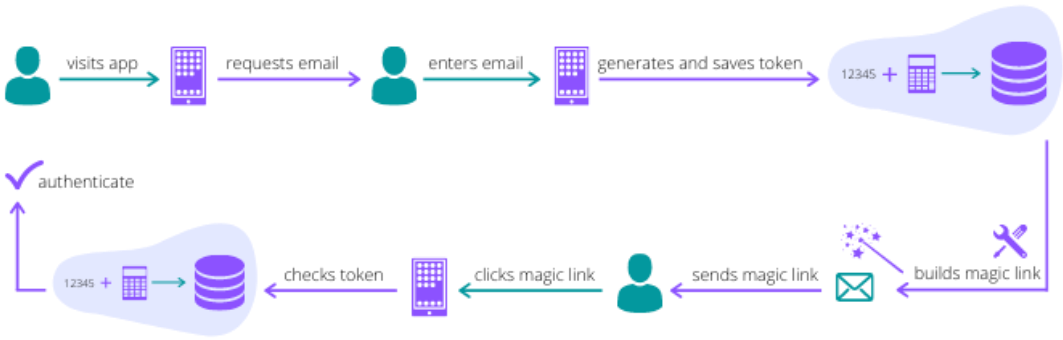

Figure 1. Magic link authentication flow [4]

The magic link's validity depends on several factors. Each link can only be used once; furthermore, there's a certain time limit, and the user cannot authenticate using an outdated magic link. Besides, the user must open the link in the same browser as the resource they are trying to gain access to.

In addition to implementing a passwordless authentication system, we have also created a 2FA system using TOTPs. To generate one-time passwords, we have used Google Authenticator – a free mobile application developed by Google which allows the user to generate TOTPs on their mobile device. To set up authentication, a QR code is used; Authenticator scans it and starts generating OTPs, creating a new OTP after each set period of time [5].

User registration and authentication using Google Authenticator are shown in Figure 2.

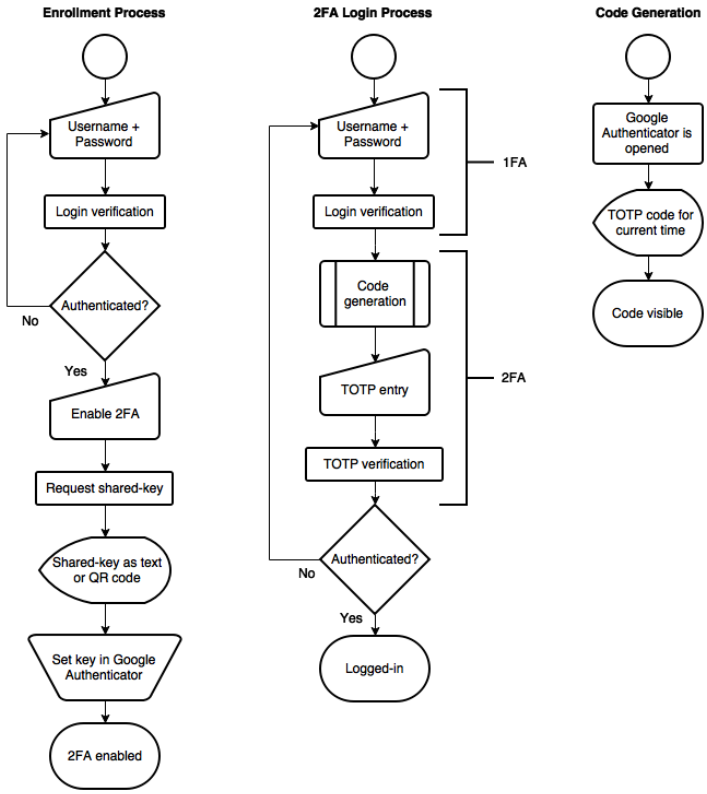

Figure 2. TOTP-based 2FA system user registration and authentication [5]

The authentication system that we have created is adaptive. This means that the system can either work as a regular password-based authentication system, or as a 2FA OTP-based system. For instance, if a user connects to the system from inside a company (that is to say, using a company IP),

they only need to enter a password; however, if they connect from the outside, they need to complete both steps of the authentication process. Such a system balances user convenience with security, based on the assumption that the connections from within a company are more secure and do not necessarily need extra protection.

4. Conclusion and Future plans

We plan to continue working on passwordless authentication technology, investigating further avenues of research in this area.

One possible direction of research is the use of the WebAuthn standard. WebAuthn is a standard for web-based user authentication which was developed by W3C and FIDO Alliance as a part of the FIDO2 protocol. It allows the users to authenticate using a variety of methods, including biometrics (inherence-based authentication) or FIDO security keys (possession-based authentication) [6]. As such, the development of that standard is an important step for the field of passwordless authentication.

Another option is to consider using decentralized identifiers (DIDs). DID is a new type of globally unique identifier, which has a number of essential characteristics: it is decentralized, meaning that there is no central issuing agency; it is persistent, not requiring the continued operation of any organization; importantly, it is cryptographically verifiable, so it is possible to prove control of the identifier using cryptographic methods [7]. There is a currently developing authentication protocol based on the decentralized identifier technology called DID Auth. In this protocol, the user authenticates by proving that they are the owner of a certain DID. The authentication is based on the "challenge-response" method, which does not transmit a secret over the communication channel.

References

[1] Firdhous, Mohamed. Implementation of Security in Distributed Systems - A Comparative Study // International Journal of Computer Information Systems, Vol. 2, No. 2, 2011

[2] Verizon Data Breach Investigations Report. Available at: https://www.verizon.com/business/resources/reports/dbir/ (accessed 05.09.2021)

[3] 2017 Consumer Mobile Security App Use. Available at: https://www.keepersecurity.com/ (accessed 05.09.2021)

[4] A guide to magic links: how they work and why you should use them. Available at: https://workos.com/blog/a-guide-to-magic-links (accessed 05.09.2021)

[5] From Theory to Practice: Adding Two-Factor Authentication to Node.js. Available at: https://auth0.com/blog/from-theory-to-practice-adding-two-factor-to-node-dot-js/ (accessed 05.09.2021)

[6] Web Authentication: An API for accessing Public Key Credentials Level 2. Available at: https://www.w3.org/TR/webauthn-2/ (accessed 05.09.2021)

[7] Decentralized Identifiers (DIDs) v1.0. Available at: https://www.w3.org/TR/did-core/ (accessed 05.09.2021)