# DESIGNING A BIOMETRIC-BASED SECURE ACCESS MECHANISM FOR CLOUD SERVICES THAT IS SECURE AND EFFECTIVE

**First Author:**

V. Chandra Sekhar, Associate Professor, Dept of MCA, ASCET, Gudur.

**Second Author:**

A. Pallavi, PG Scholar, MCA, ASCET, Gudur.

## ABSTRACT:

Safe access to such data and services is necessary due to the rising demand for remote data storage and compute services. Here, we provide a brand-new method of biometric authentication for securing access to a remote server (in the cloud). In the proposed method, biometric information is used as a secure credential. We can then produce a distinctive identification using the user's biometric information from which the private key can be generated. We come up with a quick technique for creating a secure message transmission session key between two participants using two biometric templates. The user's private key does not need to be kept on file, and the session key is generated with no prior knowledge exchanged. employing the widely used Automated Validation of Internet Security Protocols and Applications for security analysis. The Real-Or-Random model and tool (AVISPA) based on formal, informal (non-mathematical), and formal security analysis demonstrate that the suggested technique may withstand a number of well-known assaults against (passive/active) adversary. Extensive tests and a comparative analysis demonstrate the efficiency and effectiveness of the suggested strategy.

## 1. INTRODUCTION

In the world we live in, cloud services are taken for granted. It is undeniably difficult to establish reliable authentication, authorization, and accounting for access to cloud services, both operationally and scientifically. Only a handful of the numerous authentication techniques that have been discussed in the literature over the years are OpenID and Kerberos [1], OAuth [2] and Kerberos [3]. There are various kinds of protocols designed to enable the secure transfer of access rights between two cooperating organisations in a distributed system. These protocols are based on the assumption that the remote server that provides authentication is a trustworthy component of the network. A user must initially connect to a faraway server. To ensure that the owner has done this, it is required to the legal right to do so. When a user accesses a server, a remote server authenticates the user, and the user authenticates the server. A remote server gives the user access to the requested services when both checks have been successful. Through the use of current authentication systems, users' credentials can be stolen and (mis)used to gain unauthorised access to a variety of services. The majority of currently used methods rely on symmetric key cryptography, which demands that several cryptographic keys be exchanged throughout the authentication procedure in

order to guarantee both security and speed. This approach adds additional overhead to the authentication processes. According to the flaws found in the protocols published by Jiang et al. [13], Althobaiti et al. [14], Xue and colleagues [15], Turkanovic and colleagues [16], Park and associates [17], Dhillon and Kalra and associates [18], Kaul and Awasthi and associates [19], and Kang and associates [20] also see Section II: It is challenging to create secure and effective authentication protocols. The purpose of this work is to provide a fast and secure authentication mechanism. We'll start by proposing a different form of password-based authentication. Finally, we demonstrate how to establish a secure connection between persons conversing throughout the authentication process without any shared (i.e., secret pre-loaded) information being available.

## 2. LITERATURE SURVEY

[1] The kerberos network authentication service (v5), C. Neuman, S. Hartman, and K. Raeburn, RFC 4120, 2005. This document replaces RFC 1510 and gives an overview and specification of Version 5 of the Kerberos protocol. It also clarifies any parts of the protocol and its intended application that call for a more thorough or understandable explanation than RFC 1510 could offer. This paper aims to give a thorough explanation of the protocol that is acceptable for implementation, along with explanations of the proper usage of protocol messages and the fields contained inside those messages.

[2] The OAuth Protocol. [Online]. Accessible at: www.oauth.net A delegation protocol that is useful for communicating authorization choices among a network of web-enabled applications and APIs is defined by the OAuth 2.0 specification. OAuth is utilised in numerous programmes, such as by offering user authentication tools. This has caused a lot of developers and API providers to utilise OAuth wrongly since they incorrectly believe that it is an authentication mechanism in and of itself. To be clear, let's state that once more: A protocol for authentication is not OAuth 2.0. The fact that OAuth is used inside of authentication protocols contributes significantly to the confusion since developers will see the OAuth components and interact with the OAuth flow and believe that they can complete user authentication by doing nothing more than using OAuth. This proves to be false as well as potentially harmful for service providers, developers, and end users This article aims to answer the query of how to construct an authentication and identification API using OAuth 2.0 as the base for various identity providers. Basically, continue reading if you're saying, "I have OAuth 2.0, and I require authentication and identification."

[3] "OpenID Protocol," [Online]. Accessible at: http://openid.net OpenID Authentication offers a means of demonstrating that an end user is in charge of an Identifier. It accomplishes this without requiring access from the relying party to end user login credentials, such as a password, or to other sensitive data, like an email address. OpenID has no central authority. Relying Parties and OpenID providers do not need to be registered or approved by a central authority. An end user is free to select the OpenID provider they want to use and can keep their Identifier if they change providers. Although the protocol doesn't require contemporary browsers or JavaScript, the authentication method works well with Date 2022-08-10 Words 544 Characters 3564 Page 1 of 2 "AJAX"-style setups. As a result, a user doesn't need to leave their present Web page in order to establish their identity with a relying party. OpenID

Authentication does not require any specialised User-Agent or other client software functionality because it merely makes regular HTTP(S) requests and answers. The use of cookies or any other particular method of Relying Party or OpenID Provider session management is not a requirement for OpenID. Extensions to User-Agents are not necessary in order to use the protocol, although they can make end user interaction simpler. Additional service types that are constructed on top of this protocol to provide a framework can be used to address the exchange of profile information or the exchange of other information not covered by this standard. With the aid of OpenID Authentication, a free, decentralised base service for portable, user-centric digital identity is made possible.

## 3. PROPOSED SYSTEM

In order to enable safe access to a remote (cloud) server, we build a new biometric-based authentication system in this article. In the suggested method, we treat a user's biometric information as a secure credential. From the user's biometric information, we then create a unique identity that is utilised to generate the user's private key. Additionally, we offer a practical method for creating a session key for secure message transmission between two conversing participants utilising two biometric templates. In other words, the user's private key does not need to be stored anywhere, and the session key is generated secretly.

We treat a user's fingerprint image as a secure credential. We create a private key from the fingerprint image, which is then used to covertly store the user's credentials in an authentication server's database. In the authentication phase, we take a fresh image of the user's biometric fingerprint, create the private key, and then encrypt the biometric information as a query. The authentication server receives the biometric data that has been queried and compares it to the data that has been stored there. The user is now prepared to access his or her service from the chosen server after successfully completing the authentication process. Mutual authentication utilising a short-term session key has been suggested between the user and authentication server as well as between the user and service server in order to achieve secure access to the service server.

We describe an efficient and reliable method to generate the session key using two fingerprint data. For message authenticity purposes, a biometric-based message authenticator is also generated. The essential contributions and advantages of the suggested strategy are outlined below.

1) A practical method for sending a user's biometric information to an authentication server through unencrypted network channels is given.

2) We suggest a method for creating an irrevocable fingerprint image directly from a revocable private key. The private key and a direct representation of the user's biometric information don't need to be kept anywhere.

3) By eliminating the need for the user's credentials to be kept on the authentication server, we overcome the drawback of conventional techniques.

4) We present a novel technique for creating session keys.

5) Each entity needs some preloaded information in a typical authentication system, adding to the overhead. To get around the requirement for hidden pre-

loaded information, we present a new technique.

6) As an alternative to the current message authentication methods (i.e., Message Authentication Code (MAC)), a message authentication mechanism is introduced.

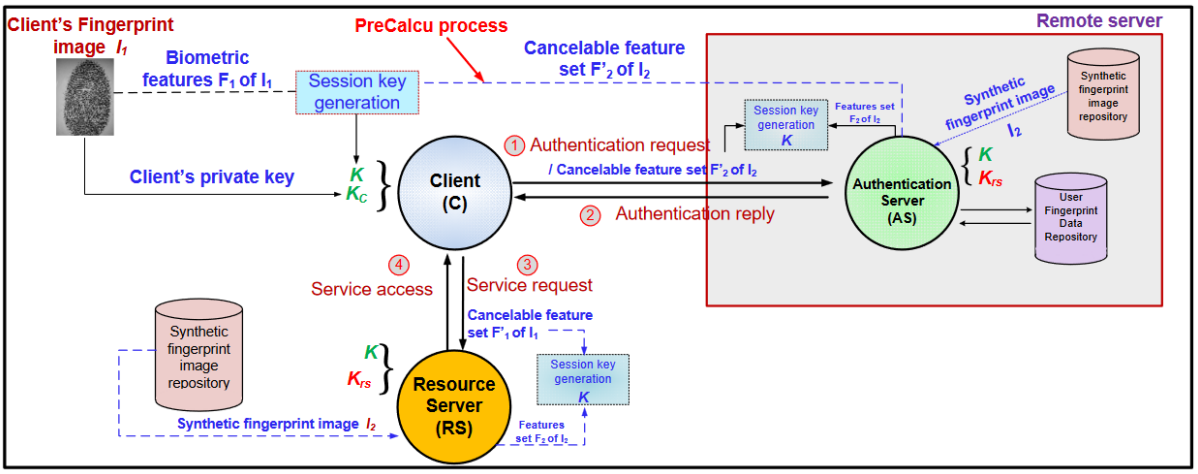


Fig: Block Diagram of Working model.

## SYSTEM MODEL

Figure 1 depicts the overall structure of the Bio CAP, which consists of three entities. These include clients, authentication servers, and possibly some resource servers (RS). Users' registration information is stored in a database on AS, which also produces RS's private key after deployment and shares it with RS. Additionally, a sizable store of fake fingerprint photos is included in both AS and RS. The suggested approach uses some synthetic fingerprint databases, such as some publicly accessible datasets.

C must first send an authentication request to AS in order to access a service from RS. When AS successfully verifies C's request, it replies to C with a message. C submits a service request to RS for access after receiving the authentication reply message. The service request is then verified by RS. When the service request is successfully confirmed, RS responds to C. Mutual authentication is performed between C and RS. For further secure message exchanges, a session key is utilised between C and AS, C and RS. A message authenticator also regulates the message's legitimacy.

The two crucial procedures in BioCAP are user registration and user authentication. While user authentication requires the production of a private key, user registration of the message authenticator and the session key. The user's private key can be rolled over via BioCAP. In addition, BioCAP overcomes the inherent limitations of biometric verification and is secure and computationally less expensive. Additionally, BioCAP doesn't require pre-shared keys, offers a seamless mutual authentication mechanism, and necessitates managing fewer keys from the application and user perspectives.

## 4. CONCLUSION

The rising use of biometric security systems shows that they have distinct advantages over traditional password and token-based security systems (e.g., on Android and iOS devices). In this research, we present an authentication method based on biometrics for users attempting to access services and computing resources from a distance. Given that a user's fingerprint may be used to produce the same key with an accuracy of 95.12%, our suggested method enables the generation of a private key from a fingerprint biometric reveal. There is no need to communicate any prior information when using the session key creation method, we suggest using two biometric data. Our strategy is more resistant to a number of known attacks when compared to other authentication methods of a similar nature.

## 5. REFERENCES

[1] C. Neuman, S. Hartman, K. Raeburn, "The kerberos network authenti- cation service (v5)," RFC 4120, 2005.
[2] "OAuth Protocol." [Online]. Available: http://www.oauth.net/
[3] "OpenID Protocol." [Online]. Available: http://openid.net/
[4] G. Wettstein, J. Grosen, and E. Rodriguez, "IDFusion: An open archi tecture for Kerberos based authorization," Proc. AFS and Kerberos Best Practices Workshop, June 2006.

[5] A. Kehne, J. Schonwalder, and H. Langendorfer, "A nonce-based protocol for multiple authentications," ACM SIGOPS Operating SystemReview, vol. 26, no. 4, pp. 84–89, 1992.
[6] B. Neuman and S. Stubblebine, "A note on the use of timestamps as nonces," Oper. Syst. Rev., vol. 27, no. 2, pp. 10–14, 1993.
[7] J. Astorga, E. Jacob, M. Huarte, and M. Higuero, "Ladon : end- to-end authorisation support for resource-deprived environments," IET Infomration Security, vol. 6, no. 2, pp. 93–101, 2012.
[8] S. Zhu, S. Setia, and S. Jajodia, "LEAP: efficient security mechanisms for large-scale distributed sensor networks," Washington D.C., USA, October 2003, pp. 62–72.
[9] A. Perrig, R. Szewczyk, D. Tygar, V. Wen, and D. Culler, "SPINS: security protocols for sensor networks," ACM Wireless Networking, vol. 8, no. 5, pp. 521–534, 2002.

[10] P. Kaijser, T. Parker, and D. Pinkas, "SESAME: The solution to security for open distributed systems," Computer Communications, vol. 1 7, no. 7, pp. 501–518, 1994.

**Author's Profile:**

Mr. V. CHANDRASEKHAR has received him MCA degree from Sri Venkateswara University in 2001, Tirupati respectively. He is dedicated to teaching field from the last 21years. He has guided P.G students. At present he is working as Associate Professor in Audisankara College of Engineering and Technology, Gudur, Tirupati (Dt), Andhra Pradesh, India.

A. PALLAVI has Pursuing his MCA from Audisankara College of Engineering and Technology (AUTONOMOUS), Gudur, affiliated to JNTUA in 2022. Andhra Pradesh, India.