IJNRD.ORG ISSN : 2456-4184
INTERNATIONAL JOURNAL OF NOVEL RESEARCH AND DEVELOPMENT (IJNRD) | IJNRD.ORG
An International Open Access, Peer-reviewed, Refereed Journal

ENCRYPTED HONEY PASSWORD AUTHENTICATION FOR ONLINE BANKING SYSTEM

[1] Reeta Joldrine A,[2]ARULMOZHI T ,[3]SWETHA T,[4] BAVITHRA A,[5]VIGNESHWARI T

[1]Assistant professor,[2345]Student Computer Science and Engineering
Sri Ramakrishna College of Engineering, perambalur, India

Abstract:Password authentication is the most widely used authentication technique, for it is available at a low cost and to deploy. Users often choose familiar words for their passwords due to the ease of remembering them.. Passwords may be leaked from weak systems. Introduces a comprehensive security framework integrating innovative techniques to enhance password protection and user authentication. The approach involves the incorporation of honey words and the implementation of the AES (Advanced Encryption Standard) algorithm for secure password storage. Augmented password-authentication key exchange (aPAKE) against insiders and honey word technique for external attackers. But none of them caresist both attacks. To address this issue, we introduce the concept of honey PAKE (HPAKE), which enables the authentication server to identify password breaches and attain a level of security surpassing traditional methods. Further, we build an HPAKE construction on the top of the honey word mechanism, honey encryption, and OPAQUE which is a standardized aPAKE. We conduct a formal security analysis of our design, ensuring resilience against insider threats and the capability to detect password breaches. We implement tour design and deploy it in the real environment. The experimental results show that our protocol only costs 71.27 ms for one complete run ,within 20.67 ms on computation and 50.6 ms on communication. This indicates that our design is both secure and viable for real-world implementation.

.

Index Terms **– Honey password, AES, TLS, Augmented password-*authentication key* exchange (aPAKE)**

I INTRODUCTION

Password authentication (PoA), which has great advantages on usability and deploy ability, is one of the most popular online authentication methods [1]. It has been attracting attentions from academia, and recently many Manuscript received 13 February2022; revised 1 September 2022;accepted 7 October 2022. Date of publication 18 October 2022; date of current version 24 February 2023. This work was supported in part by the National Key Research and Development Program of China under Grant2020YFB1805400, in part by the National Natural Science Foundation of China under Grant 62072010, in part by the China Post-Doctoral Science Foundation under Grant 2021M700215, in part by the European Union's Horizon 2020 Research and Innovation Program (ASSURED) under Grant952697, in part by the Digital Technologies Acting as a Gatekeeper to Information and Data Flows (IRIS) under Grant 101021727, in part by the Artificial Intelligence Threat Reporting and Incident Response System(TANGO) under This work was made possible through the support of Grant 101070052, supplemented in part by the High-Performance Computing Platform at Peking University.. The associate editor coordinating the review of this manuscript and approving it for publication was Prof .Deb deep Mukhopadhyay. (Corresponding authors: Wenting Li; Ping Wang.)Wenting Li is with the School of Computer Science, Peking University ,Beijing 100871, China (e-mail: wentingli@pku.edu.cn).Ping Wang is with the National Engineering Research Center for Software Engineering, Beijing 100871, China, and also with the Key Laboratory of High Confidence Software Technologies, Ministry of Education, and the School of Software & Microelectronics, Peking University, Beijing 100871, China(e-mail: pwang@pku.edu.cn).Kaitai Liang is with the Department of Intelligent Systems, Delft University of Technology, 2628 Delft, The Netherlands (e-mail: kaitai.liang@tudelft.nl).Digital Object Identifier 10.1109/TIFS.2022.3214729.

II RELATED WORK

The idea of HPAKE is natural, but how to design a concrete and secure HPAKE is full of challenges. The first one is that the ways of handling passwords between honey word mechanism and aPAKE are different. For the former method, the password plaintext needs to be transmitted to the server for verification, whereas the latter method ensures that the password remains on the client side without being sent elsewhere. One possible solution is to run multiple aPAKE instances simultaneously. Specifically, the authentication server executes t aPAKE instances, and each of them uses one sweet word; the client also run t instances, but all of them use its (real) password. For an aPAKE with explicit authentication, only the instance where the client password is equal to the server sweet word will yield a session key. But this approach increases the computational and communication cost by

the number of sweet words. Furthermore, it decreases the resistance against online password guessing by a factor oft: an attacker (without the password file) can run an HPAKE instance to verify the correctness of t password guesses (i.e. running the t aPAKE instances with the t password guesses, as the server, to interact with the client) Password authentication

While password authentication is the most common way to confirm a user's identity, it isn't even close to the most effective or secures method. Unauthorized access to your account is a real risk with weak passwords, as individuals with your credentials could gain entry without your consent, and the system lacks mechanisms to prevent it. Additionally, with the advancement of hacking techniques, weak passwords are becoming increasingly vulnerable, susceptible to being breached in shorter periods of time.

2.1 Email authentication

The user clicks the login button. This opens a mail to link that directs the person pre-written email that includes an encrypted token. **The user sends the email.** The message already comes with a recipient address so the user doesn't need to enter any information. **The server verifies the request.** Using a combination of token-based security checks, the user's identity is verified.

2.2 Biometric authentication

Biometric authentication includes any type of authentication method that requires a user's biology. While this may seem like new-age technology, you're probably already using it to unlock the screen on your smart phone. Fingerprint scanning is the most well-known form of biometric authentication, but face recognition tools are an increasingly popular choice for developers. Of course, hackers have a much more difficult time replicating a users' biological characteristics, but it is important to note that these authentication processes are often less secure than you'd initially assume. Small fingerprint scanners on smart phones only record portions of your fingerprint, for instance. A complete, clear fingerprint image offers stronger security than fragmented ones.

III LITERATURE SURVEY

3.1 PASSWORD GUESSING VIA REPRESENTATION LEARNING

We introduce a novel approach to password guessing using deep generative model representation learning. By leveraging abstract password representations, we demonstrate the versatility and effectiveness of this method, which offers new avenues for exploration in the field of password guessing. Specifically, we employ two instances of generative models: a Generative Adversarial Networks (GANs) generator and a Wasserstein Auto-Encoder (WAE). These models create smooth representations of passwords, promoting a semantic organization within the complex password space.

3.2 LEET USAGE AND ITS EFFECT ON PASSWORD SECURITY

We present a systematic approach to study the presence of Leet (or "1337") in passwords, which involves defining single and pattern forms of Leet and proposing a matching approach to detect Leet in user passwords. Single Leet forms involve substituting individual characters with Leet equivalents (e.g., "l" becomes "1" or "e" becomes "3"). Pattern Leet forms involve replacing entire patterns of characters (e.g., "leet" becomes "1337"). To check whether a user password contains Leet, we propose a matching approach that compares the password against a set of Leet patterns. We identify the most prevalent counterpart pairs of Leet manifestations by analyzing a large dataset of passwords. Furthermore, we investigate the impact of Leet in passwords by integrating Leet transformations into the probabilistic context-free grammar (PCFG) method for password cracking. We propose a method to discover both forms of Leet usage in passwords, taking into account the length of patterns by selecting an appropriate threshold for n-grams. To distinguish possible Leet changes, a password string is initially segmented into meaningful units and other remaining segments. It's important to note that passwords are influenced by individual choices, native language, and environment. Therefore, common dictionaries may not be sufficient to cover all Leet variations. Additional linguistic and contextual analysis may be required to enhance detection accuracy.

3.3 UNIVERSALLY COMPOSABLE RELAXED PASSWORD AUTHENTICATED KEY EXCHANGE.

Implemented a protocols for password authenticated key exchange (PAKE) allow two parties who share only a weak password to agree on a cryptographically strong key by communicating over an insecure network. PAKE protocols have been studied extensively in the cryptographic literature and are compelling given the widespread use of passwords for authentication.

3.4 USING AMNESIA TO DETECT CREDENTIAL DATABASE BREACHES.

We have devised the inaugural algorithm utilizing honey words to enable a target site to detect breaches in its password database without necessitating any secret persistent state. This innovation marks a significant advancement in security measures. We evaluate the efficacy of this design by employing probabilistic model checking to quantitatively assess the level of security it provides... Firstly, we demonstrate that honey words have the capability to identify a breach in a target's database without requiring any persistent secret state at the target. This finding is particularly remarkable considering prior research in the field.. Specifically, consider a threat model in which the target is breached passively but completely and potentially repeatedly. Even without the necessity of concealing information from potential attackers, Amnesia offers the capability for targets to probabilistically detect breaches autonomously. The advantages of this approach are quantified through probabilistic model checking, showcasing its efficacy in enhancing security measures.

3.5 PRACTICAL THRESHOLD MULTI-FACTOR AUTHENTICATION

We present a novel authentication concept termed (t, n) threshold Multi-Factor Authentication (MFA), empowering users to select actively t factors out of n according to their preference. Central to our T-MFAKE (Threshold Multi-Factor Authentication Key Exchange) protocol is the Threshold Oblivious Pseudorandom Function (TOPRF). However, our protocol necessitates a specific variant of TOPRF rather than its direct utilization. This tailored approach ensures enhanced security and flexibility in authentication procedures.. Specifically, in the variant with n parties, running PRF needs a fixed party and arbitrary t − 1 parties from the rest. Access structure is generalized from the notion of threshold secret sharing. In the access structure, the party combinations for secret reconstruction can be freely specified. The only requirement is monotonicity, i.e., if a combination A can construct the secret, then any combination B including A can construct the secret. With an access structure scheme, setting the required party combinations can naturally construct a variant of TOPRF we need.

3.6 SMAKA:SECURE MANY-TO-MANY AUTHENTICATION AND KEY AGREEMENT SCHEME FOR VEHICULAR NETWORK

We propose a many-to-many authentication and key agreement scheme for secure authentication between multiple vehicles and CSPs. This scheme ensures prevention of unauthorized access and provides SK-security even in cases where temporary information is leaked. To enhance efficiency, the CSP can optimize service by broadcasting an anonymous message periodically instead of generating a unique message for each vehicle. Likewise, when a vehicle seeks services from multiple CSPs, it only needs to send a single request message instead of multiple ones. This streamlines the communication process and improves overall system performance.. The many-to-many authentication and key agreement system based on multi-cloud service providers are shown in Figure 2. The system architecture comprises a registration authority (RA), multiple cloud service providers (CSPs), base stations (BSs), and numerous vehicles. The first phase is system initialization, in which the RA allocates public parameters to the system.

IV EXISTING SYSTEM

In existing system, honey word technique is proposed to detect the password leakage for the most common password-only authentication systems, password-over-TLS. This approach associates t − 1 decoy and plausible-looking passwords (i.e., honey words) to each account. Augmented password authentication key exchange (aPAKE) protocols facilitate the establishment of a session key between a client and a server, leveraging a password as the basis for authentication. Password less authentication or multi-factor authentication systems make good use of other factors, e.g., smart phone and fingerprint.

Disadvantage

- client to send the password plaintext to the server otherwise The server cannot tell if the login password is real.
- Insider attacker.
- In aPAKE, the server has to store the verifiers in the password file for authentication.
- An external attacker**.**

V PROPOSED SYSTEM

The proposed system is a security framework designed to fortify the authentication and password protection mechanisms within online banking applications. It introduces a multi-layered approach to address the escalating challenges of cyber security. Firstly, the password storage mechanism incorporates honey words, creating deceptive decoy passwords alongside real ones. This adds an extra layer of complexity for potential attackers, enhancing the overall security posture. Real passwords are further safeguarded through AES encryption, a robust algorithm providing a formidable defense against unauthorized access. The user login process is fortified by a multi-factor authentication system, encompassing traditional login factors, and honey password verification. Apivotal aspect of the proposed system involves the server's generation and transmission of decryption keys during login. This secure key retrieval allows the client to decrypt the stored password, ensuring the confidentiality of sensitive information. To augment breach detection, honey words are integrated into the verification process.

Advantage

- If an attacker gains access to the password database, deciphering the actual passwords becomes an intricate task.
- The server-based generation and transmission of decryption keys during the login process ensure secure password retrieval.
- It provides alerts for potential security breaches

VI RESEARCH METHODOLOGY

The methodology section outline the Plan and method that how the study is conducted.

5.1 Encrypted Honey Password

In this module, introducing honey words as decoy passwords alongside the real password adds a layer of deception. Honey words add an element of deception, helping to detect and deter potential attackers. After that, encrypting the honey password using the AES algorithm enhances security. AES is a widely adopted and robust encryption standard, ensuring that even if unauthorized access occurs, deciphering the encrypted passwords remains a formidable challenge.

5.2 Password verification

The Encrypted Honey Password Verification module plays a pivotal role in confirming user identity during login. It decrypts the stored password using the received decryption key from the server. Simultaneously, it checks for the presence of honey words, triggering alerts for potential security breaches. This module works in conjunction with the User Authentication module to grant or deny access based on the verification outcome.

VII MODULE DISCRIPTION

7.1 Banking Framework

Traditional authentication methods like passwords and tokens are vulnerable to modern attacks. To enhance security and mitigate these threats, let's design an interface for online banking transactions utilizing advanced password storage techniques. This module will include interfaces for both administrators and users. Administrators can view user details, account information, and more, while users can conduct various operations such as net banking, credit card transactions, and debit card transactions

.7.2 Password Register

This module explains about the user process. User has to create account to access online transaction application. User should enter the required fields for registration such as first name, address, account details, user name, password and honey password.

7.3 User Authentication

User Authentication module is responsible for verifying the identity of users during the login process .During the login process, the server provides a decryption key to the client. This key is essential for decrypting the stored password.

7.4 Transaction Process

Upon successful verification, users gain access to the banking application. This ensures that only authenticated users with the correct decryption key and honey words can proceed

VIII CONCLUTION

Developed a multi-layered security framework for online banking, incorporating advanced techniques. It utilized honey words, AES encryption, and robust user authentication, represents a significant stride towards fortifying the integrity and confidentiality of user data. The utilization of honey words provides an ingenious deception layer and AES encryption ensures the secure storage of passwords.

IX FUTURE ENHANCEMENT

Future work of this project is to propose an android based application for banking process also implement high secure measurements using Digital PIN based authentication or Bright Pass based authentication. Also have plan to improve more security to the system with low computation time and also this have been develop in android application for mobile based social network access.

X RESULT

Developed a multi-layered security framework for online banking, incorporating advanced techniques. It utilized honey words, AES encryption, and robust user authentication, represents a significant stride towards fortifying the integrity and confidentiality of user data. The utilization of honey words provides an ingenious deception layer and AES encryption ensures the secure storage of passwords.

XI REFERENCES

[1]J. Bonneau, C. Herley, P. C. V. Oorschot, and F. Stajano, “The questto replace passwords: A framework for comparative evaluation of web authentication schemes,” in Proc. IEEE Symp. Secur. Privacy, May 2012,

[2]N. Huaman, S. Amft, M. Oltrogge, Y. Acar, and S. Fahl, “They would do better if they worked together: The case of interaction problems between password managers and websites,” in Proc. IEEE Symp. Secur. Privacy (SP), May 2021, pp. 1367–1381.

[3] D. Pasquini, A. Gangwal, G. Ateniese, M. Bernaschi, and M. Conti, “Improving password guessing via representation learning,” in Proc.IEEE Symp. Secur. Privacy (SP), May 2021, pp. 265–282.

[4] W. Li and J. Zeng, “Leet usage and its effect on password security, ”IEEE Trans. Inf. Forensics Security, vol. 16, pp. 2130–2143, 2021.

[5] Have I Been PWNED. Accessed: Aug. 15, 2021. [Online]. Available: https://haveibeenpwned.com

[6] Yahoo! Data Breaches. Accessed: Aug. 15, 2021. [Online]. Available: https://en.wikipedia.org/wiki/Yahoo!_data_breaches

[7] Yahoo Tries to Settle 3-Billion-Account Data Breach With $118
Million Payout. Accessed: Aug. 15, 2021. [Online]. Available: https://arstechnica.com/tech-policy/2019/04/yahoo-tries-to-settle-3-billion-account-data-breach-with-118-million-payout/

[8] sZ.Whittaker.(2018).Github Says Bug Exposed Some Plaintext Pass-words. [Online]. Available: https://www.zdnet.com/article/github-says-bug-exposed-account-passwords/
[9] S. M. Bellovin and M. Merritt, "Augmented encrypted key exchange: A password-based protocol secure against dictionary attacks and password file compromise," in Proc. ACM CCS, 1993, pp. 244–250.
[10] V. Boyko, P. MacKenzie, and S. Patel, "Provably secure password-authenticated key exchange using Diffie–Hellman," in Proc. EURO-CRYPT. Cham, Switzerland: Springer, 2000, pp. 156–171.
[11] C. Gentry, P. MacKenzie, and Z. Ramzan, "A method for making password-based key exchange resilient to server compromise," in Proc. CRYPTO Cham, Switzerland: Springer, 2006, pp. 142–159.
[12] S. Jarecki, H. Krawczyk, and J. Xu, "OPAQUE: An asymmetric PAKE protocol secure against pre-computation attacks," in Proc. EUROCRYPT. Cham, Switzerland: Springer, 2018, pp. 456–486.
[13] M. Abdalla, M. Barbosa, T. Bradley, S. Jarecki, J. Katz, and J. Xu, "Universally composable relaxed password authenticated key exchange," in Proc. CRYPTO. Cham, Switzerland: Springer, 2020, pp. 278–307.
[14] S. Smyshlyaev, N. Sullivan, and A. Melnikov. (2020). [CFRG] Results of the PAKE Selection Process. [Online]. Available: https://mailarchive.ietf.org/arch/msg/cfrg/LKbwodpa5yXo6VuNDU66vt_Aca8/
[15] A. Juels and R. L. Rivest, "Honey words: Making password-cracking detectable," in Proc. ACM CCS, 2013, pp. 145–160.
[16] D. Wang, H. Cheng, P. Wang, J. Yan, and X. Huang, "A security analysis of honey words," in Proc. Netw. Distrib. Syst. Secur. Symp.,2018, pp. 1–18.
[17] Akshima, D. Chang, A. Goel, S. Mishra, and S. K. Sanadhya, "Generation of secure and reliable honey words, preventing false detection," IEEE Trans. Depend. Secure Comput., vol. 16, no. 5, pp. 757–769, Sep. 2019.
[18] K. C. Wang and M. K. Reiter, "Using amnesia to detect credential database breaches," in Proc. USENIX Secur., 2021, pp. 839–855.
[19] Passwordless Authentication | Duo Security. Accessed: Aug. 15, 2021.[Online]. Available: https://duo.com/solutions/passwordless
[20] W. Li, H. Cheng, P. Wang, and K. Liang, "Practical threshold multi-factor authentication," IEEE Trans. Inf. Forensics Security, vol. 16,pp. 3573–3588, 2021.