Nile Journals

Nile Journal of Communication & Computer Science

Volume 5,June 2023

Journal Webpage: https://njccs.journals.ekb.eg

Print ISSN: 2805-2366
Online ISSN: 2805-2374

Honey Algorithm for Securing and Identifying Hackers in a Pervasive Environment

Yusuf, Musa [1] , FAKI, Ageebee Silas [1] , Adelaiye, and Ishaya Oluwasegun [1]

1: Bingham University Faculty of Science and Technology Department of Computer Science KM 25 Keffi-Abuja Express Way

Abstract

The emergence of the pervasive device has made log-in details more vulnerable to unauthorized access and damage. This is due to frequent changes in users of pervasive devices and the close affinity of many attackers. Most models available only prevent attackers from gaining access to user login details. This study proposed a model that both detects and reveals the attacker's identity using the strength of the Honey Encryption algorithm with the ability to build a randomized message encoding called a Distribution-Transforming Encoder (DTE). The proposed model has the capability of providing a guide to security operatives to track and arrest the suspected perpetrator. An evaluation of the model was carried out which shows a 62% success of revealing attackers. A further examination of the model shows that 21% of the attackers could gain access through close affinity to log-in users. An extension of the proposed model can be achieved by improving the detection rate of the model.

Keywords–cryptography, Honey Encryption, pervasive, hackers, security.

1. INTRODUCTION

The intensity and capabilities of brute force attacks are on the rise meanwhile the quest for transacting business anywhere, anytime is increasing by the day. The business environment is moving away from desktop personal computers to a computing paradigm where computers are embedded in everyday artifacts like cars, home appliances, furniture, and, buildings [1]. The natural desire for the use of computing devices in our daily life is on the increase. This in turn is leading the recent technological advancement which is revolutionizing computer and ICT research with gains in pervasive networking and computing. Pervasive computing is affecting our daily life from the use of computerized bread toasters to nano-sized chips. Amid the beauty of this computing paradigm, there exist numerous ugly faces when it comes to protecting privacy [2]. The paradigm shift from one –one computer to one-to-many has thrown up a new usability challenge in login security, therefore; the onus of building better, secured and safer encryption algorithms are now the job of modern computer security experts. Though typing-in passwords and usernames which in most cases is a first-line security in these devices is straightforward but is prone to brute force attacks due to user carelessness or device usage. According to [3], many researchers have tried to bridge the gap between pervasive computing and privacy, but because

of the complexity of this system, including dimensions of security, context, and locality can only improve it.

Beyond classical security concepts like the use of firewalls to protect devices and users against intrusion or roles which rely on management metrics to define roles, rights, and context, there exist other complex security mechanisms that use logical schemes like contract-based security, RFID-based authentication, and identification and statistical methods based on neural networks [4]. Security in a pervasive environment is a big issue that makes privacy and security very elusive. This is coupled with the fact that, in most cases, ISPs do not comply with privacy and security standards set, thereby leaking subscribers' private information to fraudsters. Users are expected to be moving in different cells having different privacy policies. Also, the use of mobile devices though personal to each individual might be shared for use between families and friends, making login credentials vulnerable. Though login credentials are always encrypted using password-based encryption, their security is inadequate in preventing brute force attacks [5].

In pervasive business applications, the relationship between users and the security of pervasive devices must be strengthened [6]. This can be achieved by using better authentication and authorization mechanisms. Also, there should be an assurance that in case of compromise, the attackers involved can be traced and identified.
Many researchers have come up with various solutions to solve the problem of brute force attacks on user logins, their solutions only concentrate on denying the attacker access to the system and device. In this study, an enhanced version of the honey encryption algorithm is implemented with the aim of both preventing unauthorized access to the attacker and revealing their identities.

2. LITERATURE REVIEW

The pervasive computing paradigm has come with a lot of benefits but is having a major downside with security. This is a common trend in distributed computing. Much research has been done on the security of passwords yet, the problem continues. Most users have formed a habit of reusing or duplicating login passwords reason being that it is easy to remember without knowing that this practice can easily compromise password security [7]. Because of [8], two different password strategies using a 3-word hash and a random letter hash are used to protect online passwords. The strategy was to help users avoid the repetition of passwords on different sites. The results obtained by the study prove that most users were interested in adopting these methods to manage their login passwords. The drawback of this strategy is that such passwords are short with a pattern created for easy recalling. This proves that an attacker can easily brute-force such passwords with a little clue.

According to [9], there are various ways an attacker obtains a user password login. It includes; a brute force attack which is a trial and error (guessing) of user logins. Dictionary attack involves breaking into servers of computers by systematically entering every word in a dictionary as a password, malware which involve a Trojan program that records the keystroke of users and sends it to an attacker, visible password which are login passwords that are written on a stickie and made visible to an attacker and phishing which is the most modern method that involves an attacker cloning a web page that collects login details. [9] Proposed using honey words to protect cyberpunks. This method has two drawbacks. First, it only protects passwords but cannot review

who the cyberpunks are/are when attacks occur. Secondly, the system is compromised in an event that the attacker finds the right password by first guess. In the business environment, there is an expected trust and reliance on security by pervasive device users. These trusts if achieved can create confidence and increase usage of such devices. [10] stated that increasing pervasive user confidence requires strong security of login details. But, because most pervasive device users are novices to the ICT world, downloading malicious applications, using easy passwords with weak security settings, and answering spam emails that compromise and leak passwords to attackers is evitable. A pervasive environment being a distributed environment is susceptible to a variety of threats mounted by intruders' as well legitimate users' misbehavior. The attention of attackers on pervasive devices is on the increase because of their valuable and rich backend data such as bank accounts, corporate organization documents, and personal health information [11] and also the high mobility of devices used in the attacks. Pervasive devices are scattered everywhere within our environment due to their processing power and mobility, which create unrestricted usage for attackers. This calls for not just developing a protective mechanism for them but there is also need for a mechanism that reveals who these attackers are.

3. METHODOLOGY

Logins credentials are created by users which most times comprise of a username and password. This could be done using either a desktop computer or a mobile device which may either be on the direct webpage or a downloaded application.

Background of Brute Force Attack

Before the study provides a solution, a background on brute force attacks is necessary for better understanding. With the mindset that the main interest of an attacker is for message recovery, given an encryption cipher C = enc(K, M) of a message M and key K (where K and M are drawn from known distributions), an attacker's goal is to recover M. The attacker tries several keys to decrypt C which results in a set of messages M1, M2, M3 … Mq. If the attacker succeeds with a key K, then, access to message M is guaranteed with a winning probability equal to the attacker's ability to pick out M from the q candidate.

Honey Encryption (HE)

The strength of HE is the ability to build a randomized message encoding called Distribution-Transforming Encoder (DTE). Encrypting a message M under HE involves a two-step procedure known as DTE-then-encrypt.

Using the two-step procedure which involves first, the DTE is applied to M to obtain a seed S. Second, the seed S is encrypted under a conventional encryption scheme enc using the key K, yielding an HE ciphertext C.

Using the H = cryptographic function,
K = key, M = message,
S = seed, R= random seed,
C = ciphertext and
$ = uniform random assignment, an instantiation of DTE –then-encrypt for HE encryption and decryption is given in the algorithm below:

Encryption
Henc(K, M)
S <- $ encode(M)
R <- $ {0, 1}n
S’ <- H(R, K)
C <- S’ ⊕ S
Return (R, C)

Decryption

Hdec(K, (R, C))
S’ <- H(R, K)
S <- C ⊕ S’
M <- decode(S)
Return M

4. Revealing Attackers Algorithm (RAA)

Suppose a user password manager database is encrypted with honey encryption and decrypts under an incorrect master password P * and expected questions q, to yield a list of fake passwords and reveal the attackers who provided answers to q. The attacker will be prevented from accessing messages M and his identity i reveled with a set μ (where μ is the set of all answers to q). With this model, an adversary can’t decrypt C even when the password is weak as seen in the algorithm that follows.

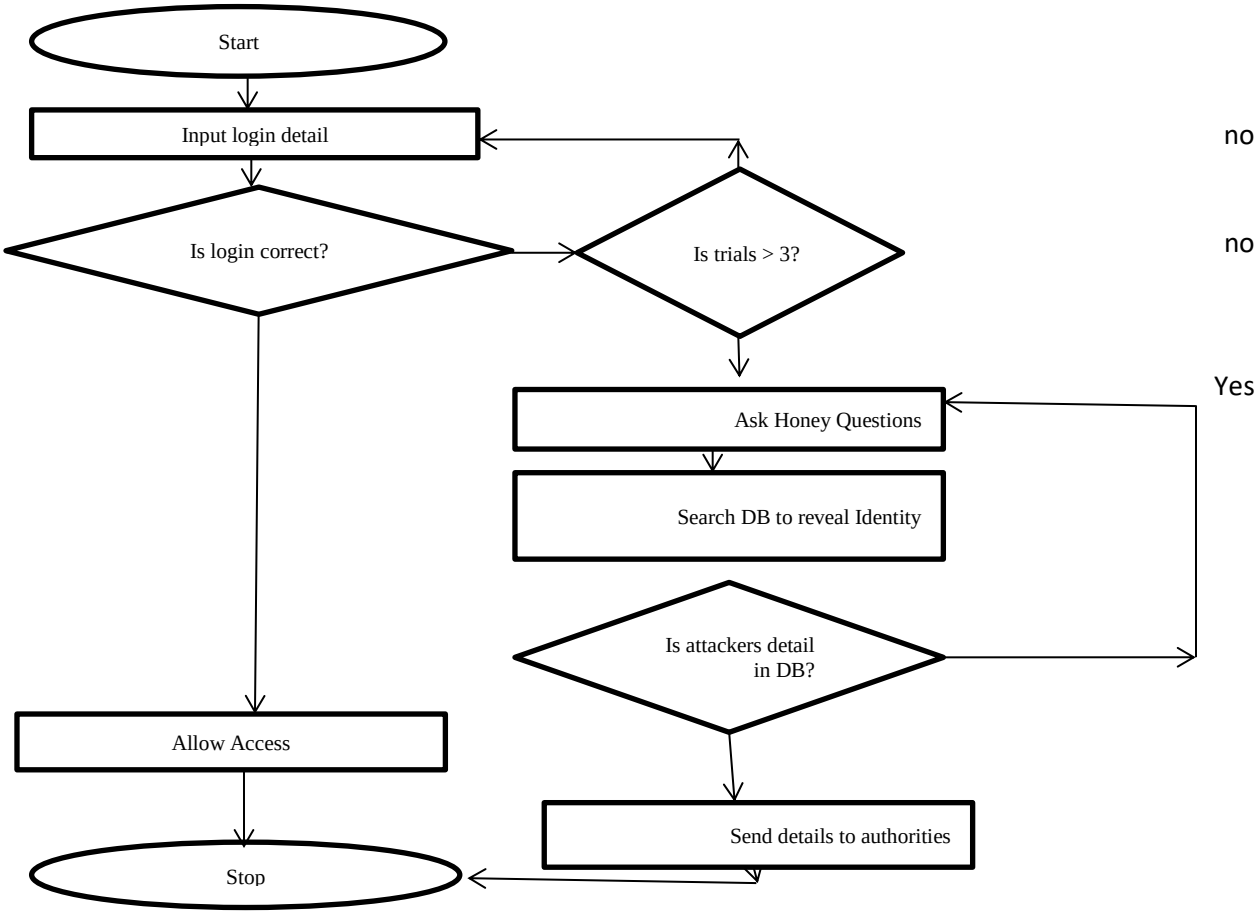

Figure I: Honey Encryption algorithm

Hence (K,M) //encrypted message M with a key K Attacker Hdec(P* ^q, M) // incorrect password P* decrypted with questions q

$\mu = \{ q1, q2, \ldots qn\}$

$i \leftarrow \mu$

The summary of the algorithm for revealing attackers is presented in the flowchart that follows.

5. EVALUATION

To evaluate the study's model, a webpage that keeps records of Computer Science students' test scores was developed and attached to the Bingham University server http://www.binghamuni.edu.ng/ with a database controlled by the lead author of this study as an administrator. All second, third, and fourth-year level students (totaling one hundred and fifty-four). were mandated to register for their continuous assessment record on the webpage. This was achieved by first creating login details (username and password) followed by biodata details. None of the students was aware that the exercise was a demo for this research. The three levels (classes) were split into four groups based on closeness with each other. Students that share the same device in their practical lab were in group one denoted by (Sd), those that are friends or related by family ties in group two denoted by (Sf), students that stay in the same hostels were in group three denoted by (Sh) and those students that have nothing in common in group four (Sn).

After registration, hypothetical marks and grades were recorded against each student. The students were then asked to brute force their group members to log in to reveal their scores. In the course of brute forcing, honey questions were asked unknowingly to the students, and answering the questions correctly reveals their identities. In a situation where students cleverly refused to give answers that reveals who they are, such students were denied access. The audit tray was monitored and information on the log-in of students was collected.

6. RESULTS OF THE EVALUATION

The combined results of each group evaluated using the study's algorithm are shown in Figure 1.

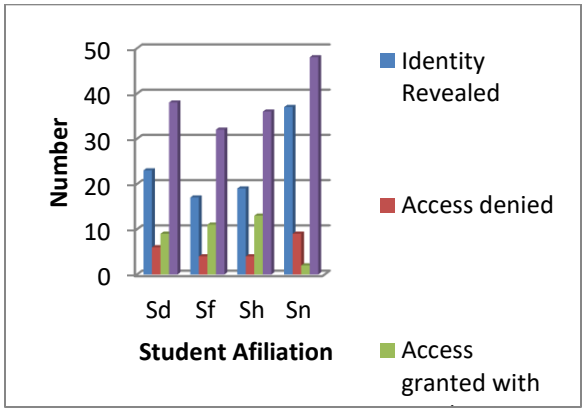

Figure 2: Result of model evaluation

From figure 2, the result of the model shows that 38 students share the same device Sd (computer) in the software laboratory. Out of this number, the model revealed 23 students' identities, 6 students declined to answer the honey questions and were denied access to the database while 9 students were able to gain access without their identity being revealed. The number of students who share family ties or were close friends Sf was 32. Out of this number, the model result showed that 17 students' identities were revealed, the model declined access for 4 students in this group while 11 students were able to gain access undetected. Also, 36 students reside in the same hostel Sh. Out of this group, 19 students' identities were revealed with 4 access denied while 13 students gained access without identify being revealed. Lastly, 46 students had no affinity concerning the three groups described before Sn. From this group, 37 students' identities were revealed and 9 accesses were denied while 2 were able to gain access undetected.

7. VI GROUP EVALUATION:

From the result obtained, the groups by group analysis were observed as:

Identity revealed: The doughnut chart shows the summary of this group.

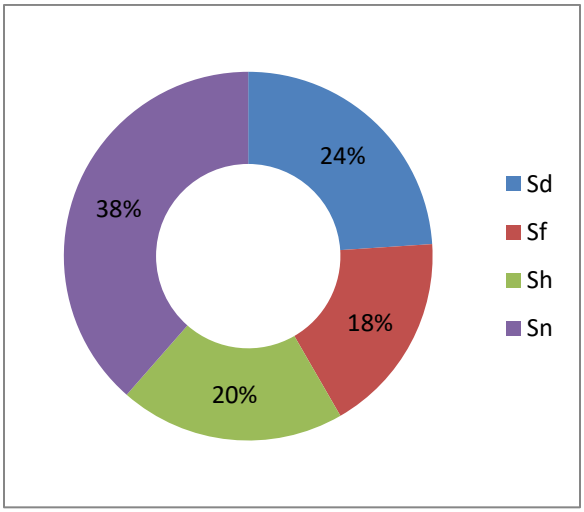

Figure 3: Number of students with revealed identity

From figure 3, it can be observed that the identity of 38% of the students who have nothing in common (Sn) was revealed, the identity of 24% of the students related by sharing the same computing device (Sd) was revealed, the identity of 20% of the students who live in same hostel (Sh) was revealed and the identity revealed by students related by friendship or family ties (Sf) was 20%. The implication of Sn having a higher percentage of revealed identity is because family members, friends, or neighbors know much about themselves and therefore can easily guess or get their login details. This is so because many people use the date of birth, pet names, or related names or events that have occurred to them as login details, which can easily be guessed by those close to them.

Accessed Denied: The doughnut chart shows the summary of this group.

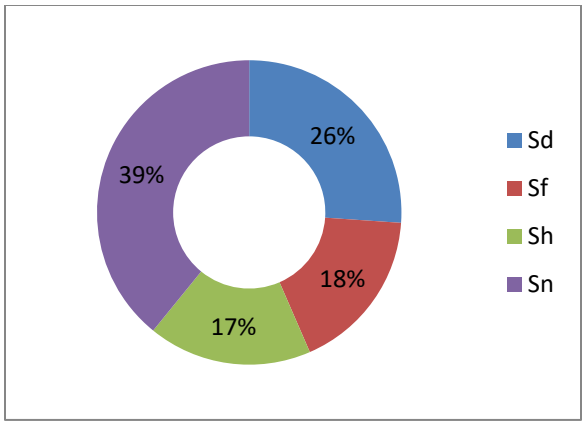

Figure 4: Number of students with access denied

Figure 4: shows that 39% of the students with nothing in common (Sn) have access denied in logging in, 26% of student who shares the same computing device (Sd) were denied logging in, 18% of students who has family Sf ties could not log in while 17% of students who live in the same hostel Sh could not gain access to the system.

Accessed granted: The doughnut chart shows the summary of this group

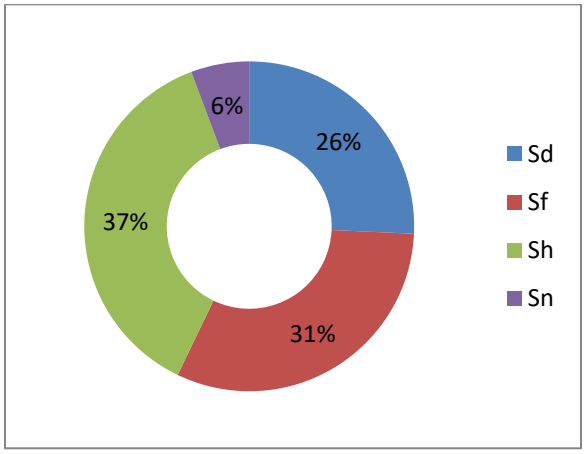

Figure 5: Access granted with no identity revealed

Figure 5 shows that 37% representing those students who reside in the same hostel could log in successfully without their identity being known, 32% of students who have links with family ties gained access without knowing who they were, 26% of students who share the same device in the laboratory were able to log in undetected while 6% of students who share nothing in common were able to log in successfully.

8. SUMMARY

From the model, a further investigation shows that students who have no affinity with each other had the least penetration. This is because relationship encourages carelessness. Also reuse of password, which is known by users' close associates are the first to be used in brute forcing. In conclusion, names such as nicknames, pet names and parents' names, date of birth are commonly used by users as login and could be easily guessed by close associates. On the whole, the model

was able to reveal and protect user logins especially, where the user is careful and had no close affinity to the attacker.

9. REFERENCES

[1] Jurgen B, Vlad C, Marc L, Friedmann M. & Michael R (2010). Living in a world of smart everyday objects—social, economic, and ethical implications, Human and Ecological risk Assessment: An international Journal.

[2] Mohammad S. O., Mieso D. & Isaac W. (2011), Pervasive computing and Networking. John Wiley & Sons Ltd, the Atrium, Southern Gate, Chichester, West Sussex, PO19 8SQ, United Kingdom,

[3] Jaydip S (2010). Ubiquitous computing: Potentials and challenges, Proceedings of the International Conference on Trends & Advances in Computation & Engineering, India

[4] Tobias H. Oliver K. (2017) Firewall functions and roles for company security, Control Engineering, CFE Media

[5] Ari J. & Thomas R. (2013). Honey Encryption,. The security beyond the brute force bound.

[6] Michael F. & Oliver R. (2010), Ubiquitous computing: An overview of technology impacts, Elsavier, Telematics and Informatics

[7] Jeff H. (2017). What can you do to protect from password reuse, SecureAuth + coreSecurity

[8] Samira, S., Santosh, V. & Adam, T. (2017). Usability of humanly computable passwords, Cornel University Library, Retrieve January 1st from arXiv.org > cs > arXiv:1712.03650

[9] Reshma N. & Shivamurthy G. (2016). An approach for password authentication using honey words, International Journal of Engineering Research and General Science, 4(3), May-June,

[10] Jordi F., Francisca H., Andrés M., Florina A., Javier L., Jose A. M., Marc L., & Daniel D. (2010) Pervasive authentication and authorization infrastructures for mobile users. Elsevier, Computer and Security, 29(4).

[11] Kelvin J. (2012). SANS Mobility/BYOD Security survey, Analyst Program, SANS whitepaper