International Journal of Scientific Research in Science, Engineering and Technology (www.ijsrset.com)
© 2020 IJSRSET | Volume 7 | Issue 1 | Print ISSN: 2395-1990 | Online ISSN : 2394-4099
DOI : https://doi.org/10.32628/IJSRSET207139

Technique to Thwart Brute-Force Attack : A Survey

Lijimol James[1], Dileesh E D[2]

[1]M. Tech Scholar, M Tech Scholar, Department of Computer Science and Engineering, GEC Idukki, Kerala, India

[2]Assistant Professor, Department of Computer Science and Engineering, GEC Idukki, Kerala, India

ABSTRACT

Most encryption techniques have one essential problem that is they are vulnerable to brute-force attacks. Techniques used to prevent brute force attacks are increasing password length, password complexity, limit login attempts, using captcha, two factor authentication etc. A new mechanism to prevent against brute-force attack was introduced and this mechanism is known as Deception Model. Deception-based security mechanisms focus on altering adversaries perception in a way that can confuse them and waste their time and resources. This technique exploit adversaries biases and present them with a plausible looking but fake data. This technique can be used to serve for encrypting e-mail messages, human-generated messages and documents that are stored at different platforms. This paper will offer an detailed study of the current situation of Deception Model.

Keywords: Deception, Encryption, Brute force attack, Honey Encryption, Decoy messages, DTE, NLP, Conventional encryption, Deep Learning

I. INTRODUCTION

Computer security is a battle field between the attackers and the defenders. The very common method used by the defenders is encryption. Encryption is used to ensure that the data is secure, even if some one else gets the data. Although encryption has one essential problem, it can be cracked with a brute-force attack. Since computational power is ever increasing, the attackers have easier time to break encrypted data and algorithms. So far, many different techniques have been made to prevent brute-force attack. One of the technique is increasing the length of the encryption key, so it takes longer time to break it. A new mechanism to prevent against brute-force attack was introduced and this mechanism is known as Deception Model.

The term deception has a history of representing something fake in a software security field. Deception is used to lure the attacker into a fake target , preventing them for getting information from the real target. The system has several features that allow you to get information about an attacker instead of providing sensitive information to them. A decoy-based deception model is used for preventing brute force attack from stealing encrypted messages. This technique produces convincing decoy messages (which are coherent, contextually correct and domain-specific) to be served to an eavesdropper attempting to decrypt transmitted message

during communication. Any key supplied by the eavesdropper during a decryption process will yield a plausible message, thus, exhausting his time and resources, unlike conventional encryption scheme which yields random gibberish upon decryption with an incorrect key. The proposed deception model does not eliminate encryption but reinforces it with a degree of deception to exhaust the attackers time and resources. The decoy messages can be generated by using natural language processing, data mining, neural networks etc.

The rest of the paper is organized as follows: background studies in section II. In section III, the survey on the various deception models is presented. Section IV describes our conclusion.

II. BACKGROUND STUDIES

A. Brute-force Attack

Brute force attack is that an attacker trying with different passwords with the hope of eventually guessing correctly. The attacker systematically checks all possible passwords until the correct one is found. During password-guessing, the method is very fast to check all short passwords, but for longer passwords other methods such as the dictionary attack are used because a brute-force search takes too long.

B. Conventional Encryption

Conventional encryption involves transforming plaintext messages into ciphertext messages that are to be decrypted only by the intended receiver. Both sender and receiver agree upon a secret key to be used during encryption and decryption.

C. Distribution Transforming Encoder (DTE)

The Honey Encryption scheme is designed with a cryptographic primitive called the Distribution Transforming Encoder (DTE). The DTE is a set of algorithm DTE= (encode, decode), where encode takes a Message Space M as an input and returns a value in the Seed Space S as output. Decode takes as input a value S and returns an output message M. Honey encryption involves a DTE-and-then-encrypt process. This means a sender applies the DTE to the original message he intends encoding and then uses any conventional encryption scheme as the second layer of encryption.

The DTE represents the model of the message. A good DTE is designed to model the message distribution well such that if a seed is selected uniformly at random and applied to it, the message is recovered. The intuition here is to make the encoding process randomized to provide proper secrecy and make the decoding process deterministic.

D. NLP

Natural language processing is a subfield of linguistic, computer science, information engineering and artificial intelligence that is concerned with the interactions between computers and human languages. In particular, it focus on how to program computers to process and analyze large amounts of natural language data. The main challenges in NLP frequently are speech recognition, natural language understanding and natural language generation.

E. Deep Learning

Deep Learning is a broader family of machine learning methods based on artificial neural

networks. Deep learning architectures such as deep neural networks, deep belief networks, recurrent neural networks and convolutional neural networks have been applied to fields including computer vision, speech recognition, natural language processing, audio recognition, social network filtering, machine translation, bioinformatics, drug design, medical image analysis, material inspection and board game programs etc.

III. CRYPTOGRAPHIC ALGORITHM

In cryptography, encryption is the process of changing a message or information (plaintext) to a meaningless and unreadable form (ciphertext) so that only authorized parties can access it and those who are not authorized cannot. In an encryption scheme, the intended information or message, referred to as plaintext, is encrypted using an encryption algorithm to generate ciphertext that can be read only if decrypted. The encryption and decryption keys are the same for symmetric encryption and the encryption and decryption keys are different for asymmetric encryption.

Ramesh Yegireddi and R Kiran Kumar [1] proposed a survey on cryptographic algorithm, to make a comparative study for most important algorithms in terms of data security effectiveness, key size, complexity and time, etc. The table 1 gives the comparison between all algorithms with respect to create by, year, key size, block size, round, structure, flexible, and features.

IV. TECHNIQUES USED TO GENERATE DECEPTION MODEL

Maya Bercovitch and Meir Renford [2] proposed Honey tokens that are artificial digital data items planted deliberately into a genuine system resource in order to detect unauthorized attempts to use information. "HoneyGen" is a novel method for generating honey tokens automatically. HoneyGen generates honey tokens that are similar to the real data by extrapolating the characteristics and properties of real data items. The honey token generation process consists of three main phases that is rule mining, honey token generation and likelihood rating. In rule mining in which various types of rules that characterize the real data are extracted from the production database; honey token generation in which based on the extracted rules an artificial relational database is generated; and the likelihood rating in which based on its similarity to the real data a score is calculated for each honeytoken.

Juels and Rivest [3] proposed honeywords (decoy passwords) to detect attacks against hashed password databases. For each user account, the legitimate password is stored with several honeywords in order to sense impersonation. If honeywords are selected properly, a cyber-attacker who steals a file of hashed passwords cannot be sure if it is the real password or a honeyword for any account. Moreover, login with a honeyword will trigger an alarm notifying the administrator about a password file breach. The paper proposes an alternative approach that selects the honeywords from existing user passwords in the system in order to provide realistic honeywords, a perfectly flat honeyword

generation method which reduce the storage cost. The generating methods are Chaffing-by-tweaking, Chaffing-with-a-password-model, Chaffing with "Tough Nuts" and Hybrid Method.

Hyun-Ju Jo and Ji Won Yoon [4] proposed two new counter measures that have recently been introduced to address the conventional countermeasures that cannot preserve secrecy against various cryptanalysis approaches, including side channel analysis and brute-force attacks. The two techniques are honey encryption and the structural code scheme. Both methods look different; however, they have similar security goals and they both use feature distribution transforming encoders based on statistical schemes. The purpose of this scheme is to create false plaintext to protect the original plaintext. First, a system user trains text data to obtain the word frequency, given an assumption that plain text follows the Markov process. The next step is to generate a corpus, which is a database for an encoding and decoding process.The final step is to encode or decode user data with the corpus.

Marc Beunardeau and Houda Ferradi [5] proposed encryption paradigm designed to produce ciphertext yielding plausible-looking but bogus plaintexts upon decryption with wrong keys. The paper proposes a new technique known as Probabilistic context free grammars (PCFG). In this technique, the sentence is converted to the syntax tree or parse tree to generate a skelton. The rules are rewritten and applied to the skelton to generate a decoy data.

Joo-Im Kim and Ji Won Yoon [6] proposed a technique that efforts to strengthen security of Instant Messaging (IM) system. Conventional message encryption uses a secret or private key. However, the key is vulnerable to a brute force attack, causing to acquire the original message. A countermeasure was introduced that is plausible looking but fake plaintexts, other than the conventional message encryption which is vulnerable to brute force attack. The technique is called HoneyEncryption (HE). Plaintext are encoded using distribution-transforming encoders (DTE) and cumulative massive function (CMF) served as a code table, converting hexadecimal numbers to characters and vice versa.

Ben Whitham [7] proposed four new designs for automating the construction of honey file content. Each of the designs employed the same three stages: template selection, content extraction, and document population. The new designs select a document from the target directory as a template and employ word transposition and substitution based on parts of speech tagging and n-grams collected from both the target directory and the surrounding file system. The new designs were able to successfully mimic the content from the target directory.

Edwin Mok [8] proposed a technique for cloud data protection know as extended honey encryption (XHE). An eXtended Honey Encryption (XHE) scheme is done by adding an additional protection mechanism on the encrypted data. When the attacker attempts to access these encrypted data by entering the incorrect password, instead of rejecting the access, the HE algorithm generates an indistinguishable bogus data, in which the attacker could not

determine whether the guessed password is working correctly or not. XHE algorithm is divided into 2-sub algorithms in order to protect the file's name and file's extension.

Table 1: Comparison on Cryptographic Algorithms

Algorithms	Key Size	Block Size	Round	Structure	Flexible	Features
DES	64 bits	64 bits	16	Fiestel	No	Not Strong Enough
3DES	112 or 168 bits	64 bits	48	Fiestel	Yes	Adequate Security and fast
RSA	1024 to 4096 bits	128 bits	1	Public Key Algorithm	No	Excellent Security and Low Speed
ECC	variable	variable	1	Public Key Algorithm	Yes	Excellent Security and fast Speed
RC6	128 to 256 bits	128 bits	20	Fiestel	Yes	Good Security
AES	124, 192, 256 bits	128 bits	10, 12,14	Substitution and Permutation	Yes	Security is excellent. It is best in security and Encryption performance

The file name is given as the input to the DTE encode1 algorithm to obtain a set of prefix seed value from seed space. Then the file extension is given as the input to the DTE encode 2 algorithm to obtain a set of suffix seed value from seed space. IS_encode1 and IS_encode2 algorithm is used to generate a series of fake file names and file extensions.

Prakruthi Karuna and Hemant Purohit [9] proposed a novel comprehensibility manipulation framework (CMF) to generate comprehend fake documents, which can be used for fooling attackers and increasing the cost of data ex-filtration by wasting their time and resources. CMF requires an original document as input and generates fake documents that are both believable and readable for the attacker, possess no important information, and are hard to comprehend. CMF for generating fake documents consists of four modules: input preparation, pre-processing, modification, and post-processing.

Abiodun Esther Omolara [10] proposed HE an encryption scheme that supplies valid-looking, but fake plaintext for every incorrect key used by an intruder to decrypt a message. All possible messages relative to the original message are mapped to a seed space such that any key supplied by the attacker

when decrypting a message produces a relative, but fake message from the original message and this makes it difficult for him to determine if he has recovered the original message or not. Decoy messages are generated by using Stanford Dependency Parser and Wordnet from Princeton.

Abiodun Esther Omolara [11] proposed decoy-based deception model for preventing eavesdroppers from stealing encrypted messages and applied it for the security of instant messaging on real-world deployment. Any key supplied by the eavesdropper during a decryption process will yield a plausible message, thus, exhausting his time and resources, unlike conventional encryption scheme which yields random gibberish upon decryption with an incorrect key. The proposed deception model uses BLSTM-RNN and HMM to generate decoy messages. The BLSTM-RNN model will be used to classify the domain (intent) to which a plaintext falls into, identify and extract important/special keywords from the plaintext. The HMM scans through the identified domain and then generate a decoy message sharing similar domain as the plaintext but without any of the special/keywords.

V. CONCLUSION

This paper looked into the challenges and solutions of Deception Model. The deception was said to provide protection against brute-force attacks. Deception-based security mechanisms focus on altering adversaries' perception in a way that can confuse them and waste their time and resources. This technique exploit adversaries biases and present them with a plausible looking but fake data. This paper presents a survey on the deception model. Survey focussed on how the decoy data's are generated and where all it can be implemented. It can serve for encrypting e-mail messages, human-generated messages and documents that are stored at different platforms.

VI. REFERENCES

[1]. Ramesh Yegireddi, R Kiran Kumar "A survey on Conventional Encryption Algorithms of Cryptography" IEEE International Conference, 2016.

[2]. Bercovitch M, Renford M, Hasson L, Shabtai A, Rokach L and Elovici Y (2011). "HoneyGen: An automated honeytokens generator", in Intelligence and Security Informatics (ISI), 2011 IEEE International Conference on, IEEE, pp. 131–136.

[3]. Imran Erguler "Achieving Flatness: Selecting the Honeywords from Existing User Passwords" IEEE Transactions on Dependable and Secure Computing, 2015

[4]. H. J. Jo and J. W. Yoon, "A new countermeasure against brute-force attacks that use high performance computers for big data analysis," Int. J. Distrib. Sensor Netw., vol. 11, no. 6, 2015, Art. no. 406915. doi: 10.1155/2015/406915.

[5]. M. Beunardeau, H.Ferradi, R.Géraud, and D.Naccache, "Honeyencryption for language-robbing Shannon to pay turing?" in Proc. Int. Conf. Cryptol. Malaysia, Springer, 2016, pp. 127–144.

[6]. K.Joo-Imand J.Yoon, "Honeychatting: "A novel instant messaging system robust to eavesdropping over communication," in Proc. IEEE Int. Conf. Acoust. Speech Signal Process. (ICASSP), Mar. 2016, pp. 2184–2188. Accessed: Jan. 12, 2019.

[7]. BenWhitham.2017. "Automating the Generation of Enticing Text Content for High-Interaction Honeyfiles". In Proceedings of the 50th Hawaii International Conference on System Sciences.

[8]. P. Karuna, H. Purohit, R. Ganesan, and S. Jajodia, "Generating hard to comprehend fake documents for defensive cyber deception," IEEE Intell. Syst.,vol.33,no.5,pp.16–25,Oct.2018 doi:10.1109/mis.2018.2877277.

[9]. Mok, E., Samsudin, A., & Tan, S.-F. (2017). "Implementing the honey encryption for securing public cloud data storage". In In Proceedings of First International Conference on Computer Science and Engineering.

[10]. Omolara,A.E.,Jantan,A.,Abiodun,O.I.,&Poston, H.E.(2018) "A novel approach for the adaptation of honey encryption to support natural language message" In Proceedings of the International MultiConference of Engineers and Computer Scientists (Vol. 1).

[11]. Abiodun Esther Omolara, Aman Jantani, Oludare Isaac Abiodun, Kemi Victoria Dada, Humaira Arshad "A Deception Model Robust to Eavesdropping Over Communication for Social Network Systems" IEEE Access 2019, Digital Object Identifier 10.1109/ACCESS.2019.2928359.

Cite this article as :

Lijimol James, Dileesh E D, "Technique to Thwart Brute-Force Attack : A Survey", International Journal of Scientific Research in Science, Engineering and Technology (IJSRSET), Online ISSN : 2394-4099, Print ISSN : 2395-1990, Volume 7 Issue 1, pp. 235-237, January-February 2020. Available at doi : https://doi.org/10.32628/IJSRSET207139
Journal URL : http://ijsrset.com/IJSRSET207139