IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

Optimized Methods Of Minimizing Hackers' Threats From Rainbow Table Attacks And Brute Force Attacks In Cyber Security For Secured Network And Information Security

[1]Dasaka VSS Subrahmanyam,

[1]Professor and Vice-Principal,

[1]Department of Computer Science and Engineering,

[1]Keshav Memorial Engineering College, Uppal, Hyderabad, Telangana, India.

Abstract: Cyber-attacks can be in many ways on computer systems, in any computer network. Even though there have been several ways of attacks, more popularly and frequently used methods of attacks are Brute Force attacks (BFA) and Rainbow Table attacks (RBTA). These two methods of attacks have very negative impact on computer networks, which breaks security and prone to cyber attacks in information security, comparing to the other remaining methods. In this paper, all possible methodologies are analyzed in order to make a secured computer and information networks, by providing security to all of its computer devises. An attempt is made to minimize (to the least) all possible threats of Hackers by Brute Force and Rainbow Table attacks in cyber security.

Index Terms - Brute Force Attacks, Cyber Security, Hash Function, Password, Rainbow Table Attacks.

I. INTRODUCTION:

The main objective of hackers' attacks is to crack passwords (to decrypt encrypted data) in order to gain control over various unauthorized accounts for various malicious and unlawful things. They use different types of attacks which include Brute Force attacks, Dictionary attacks, Rainbow Table attacks, Password Spraying, Credential Stuffing, Phishing, Man-in-the-Middle attack and so on. The most commonly and frequently used methods are Brute Force and Rainbow Table attacks. If proper care were taken more than 80% (approximately) of attacks can be prevented from users' side. Here in this paper, the main focus will be on attacks of type Brute Force and Rainbow Table.

Rainbow Table is a Hash Table that contains Hash Functional values. All these hash values are unique by themselves and there is a one-to-one mapping between passwords and hash functional values in the Rainbow Table. Once a password is entered by an user, that password will be transformed into a hash functional value by using Quadratic hashing and Double hashing functions and that functional value is stored into the RBT. These values are not stored in textual format but stored in several cells of the corresponding hash function. Reversing the concept, if a hash functional value, in RBT, is cracked by attacks, then the corresponding password will be known very easily. That is why RBT is used to crack users' passwords through Hash Table. RBT is used to crack hash functional values (of the corresponding passwords) which have been generated by using several hashing algorithms such as MD5, NTLM, SHA-1 and so on. All these algorithms provide the actual plain text of a hash function of the corresponding password.

Hash functions produce various Hash Functional values depending upon the physical nature of passwords generated by users. Simple passwords provide simple Hash functional values and complex nature of passwords generate complex hash functional values through hash functions in the RBT. Cracking of Hash functional values depending on the complex structure of passwords. It is not possible to crack all passwords but some passwords of weak nature can be cracked. The first fundamental step in cyber security should begin with the creation of stronger and strongest passwords. This paper focuses on generating stronger passwords to protect users from cyber-attacks. Hackers use different kinds of algorithms to produce plain texts of hash functional values as once a hash functional value is generated for a password, then its textual representation will automatically be vanished. That is why hackers use RBT to get textual representations of passwords. One more advantage of RBT is that it provides the corresponding password without any computational processes on hash functional values.

MD5 (Message-Digest algorithm) (previously it was MD4 algorithm) algorithm is a cryptographic protocol which is used for content verification. And also uses for authenticating messages as well. It is also used for recognizing digital signatures of the corresponding hash functional values. This algorithm verifies the file sent by the user and compares the two files. Once these two files are matched, then hash functional value will be recognized and the corresponding password will be cracked. It is a most widely used algorithm among software industries which provides a minimum guarantee of cracking passwords. The degree of cracking hash functional values varies from one to another one in RBT. File servers provide a pre-calculated MD5 checksum for the files. Then, the user compares the checksum of the file sent by the server. If these two are matching each other, then the hash functional value in turn the corresponding password will be cracked. Password strength always be taken into consideration. Its time complexity is very high if stronger passwords are generated.

The next commonly used algorithm is SHA-1 (Secure Hash Algorithm) which accepts an input and generates a 160-bit (20-byte) Hash functional value. This value is known as message digest. This value will be converted into hexadecimal number which is of 40-digit length. It was practiced by US National Security Agency. It had many drawbacks in its implementation. All major companies stopped its usage.

Microsoft has been using NTLM algorithm in its Windows new technology. It provides authentication identity and protects the integrity of users' activities. It is a tool that depends upon challenge-response protocol to confirm users without requiring them to submit a password. Now this technology is replaced by Kerberos. It acts as a default authentication protocol in Windows 2000 and in other related domains. This algorithm works on the basis of the following:

Negotiation Message: from the client
Challenge Message: from the server
Authentication Message: from the client

Generation of all messages is based upon the strength of the passwords created by users. For stronger/strongest passwords, generation of messages will be complex, and even if generated, Negotiation message from the client and Challenge message from the server can't be communicate each other. Thus, if users are able to generate stronger passwords, then time complexity of this algorithm is very high.

It is clearly understood that all these password crack algorithms face very difficult complex computations, if users use very stronger passwords for their systems.

Brute Force algorithm is used to decrypt the encrypted data used by users i.e., to crack passwords by direct attacks. In majority of cases, users generally won't give importance to their passwords. It makes a great opportunity to hakers to acquire users accounts unauthorizedly. This algorithm works accurately on weak passwords. It can't work on stronger passwords. Some commonly used passwords by users are: 1234, 4444, 99999, 1000, 24680, 13579, bujji, iloveu, rose143, sweetheart, darling, chanti, nani, 1947,000, 15847 etc., Hakers use these kinds of passwords in their first attempt of attacks on users' passwords. Thus, the preliminary step in getting security and to protect their systems from hackers is to have secured passwords. Once stronger passwords are generated, then systems will be in protected domains even though various kinds of attacks were done. In Brute Force method, hackers use all possible popular text phrases as well as collected simple quotations from web sites.

Recently, it is found in surveys hat still more than 52% of users are not serious about their passwords. They are not taking any serious measures to protect their systems. It might be because of not aware of the resultant serious consequences that will happen. Every user should be aware of the serious consequences that will affect their organizations, as they cripple the entire networks of their working places that may collapse the entire information system. A computer network system consists of many sub-set systems which may further divided into many individual systems. If any single individual system gets affected, then the entire network system might be inflicted Worsley. Thus, more cautions are to be injected into the flow of network protection, which has always been a dynamic flow.

In order to protect computer and information networks from RBT and Brute Force attacks, all users, in an organization, have to follow the strict password generation procedures. There are four different kinds of sets i.e., set 1) 26 upper case English alphabets, set 2) 26 lowercase English alphabets, set 3) 10 numeric digits from 0 to 9, and set 4) 32 special characters, such as @, #, $, %, ^, &, etc., on any computer key board. Strength of passwords depend upon the length of the password and using at least one character from each of the set. Generation of hash function is very easy for passwords like 1234 which are of homogeneous nature. It is strictly advised to include heterogeneous characters of all 4 sets in any password. The main purpose of this attempt is to increase the size of the set exponentially, then it will be an ocean of searching options, for Hackers. Thus, their attacks will be a futile venture. Here, various kinds of possibilities with the varying lengths of passwords are to be examined, mathematically.

Mathematical descriptions of generating all possible choices of passwords:
Here, all the required passwords are to be generated from the available four sets of characters by considering various combinations of characters i.e.,
Number of Uppercase Letters = 26 -------- Set A ---------- (1.1)
Number of Lowercase Letters = 26 -------- Set B ---------- (1.2)
Number of Numerical Digits = 10 -------- Set C ---------- (1.3)
Number of Special Characters = 32 -------- Set D ---------- (1.4)
Let it be started with a small size of length 3. Then it will be sequentially incremented.

- For strings of length 3:
  if it were from Set A: any 3 characters can be selected in 26C3 (where C is a combinatorial computation) = (26 x 25 x 24)/(3 x 2 x 1) = 2600 ways.
  If it were from Set B: any 3 characters can be selected in 26C3 ways = 2600 ways. If it were from Set C: any 3 characters can be selected in 10C3 ways = (10 x 9 x 8)/(3 x 2 1) = 120 ways.
  If it were from Set D: any 3 characters can be selected in 32C3 ways = (32 x 31 x 30)/(3 x 2 x 1) = 4960 ways.
  If each character is selected from Sets A,B,C respectively, then, the number of ways of choosing passwords = 26C1 x 26C1 x 10C1 = 26 x 26 x 10 = 6760 ways.
  If each character is selected from Sets B,C,D respectively, then, the number of ways of choosing passwords = 26C1 x 10C1 x 32C1= 26 x 10 x 32= 8320 ways. etc.,
  It is shown that more number of passwords are created by choosing characters from different sets.
- For strings of length 4:
  if it were from Set A: any 4 characters can be selected in 26C4 ways = 14950 ways.
  If it were from Set B: any 4 characters can be selected in 26C4 ways = 14950 ways.
  If it were from Set C: any 4 characters can be selected in 10C4 ways = 210 ways.
  If it were from Set D: any 4 characters can be selected in 32C4 ways = 35960 ways.
  If each character is selected from each of the Sets A, B, C, D respectively, then, the number of ways of choosing passwords = 26C1 x 26C1 x 10C1 x 32C1 = 26 x 26 x 10 x 32 = 216320 ways.
  It is very clear by the above calculations, by comparing the number ways of creating passwords, that choosing one character from each of the Sets gives 2,16,320 ways.
- For strings of length 5:
  if it were from Set A: any 5 characters can be selected in 26C5 ways = 65780 ways.
  If it were from Set B: any 5 characters can be selected in 26C5 ways = 65780 ways.
  If it were from Set C: any 5 characters can be selected in 10C5 ways = 252 ways.
  If it were from Set D: any 5 characters can be selected in 32C5 ways = 201376 ways.

If each character is selected from each of the Sets A, B, C, D respectively, then, the 5 th character can be from any of the 4 sets. Then the possible ways of selecting the character are : A,B,C,D,A or A, B, C, D, B or A, B, C, D, C or A, B, C, D, D .

If it were from A, B, C, D, A: number of ways of choosing passwords = 26C1 x 26C1 x 10C1 x 32C1 x 26C1= 26 x 26 x 10 x 32 x 26 = 5624320 ways.

If it were from A, B, C, D, B: number of ways of choosing passwords = 26C1 x 26C1 x 10C1 x 32C1 x 26C1= 26 x 26 x 10 x 32 x 26 = 5624320 ways.

If it were from A, B, C, D, C: number of ways of choosing passwords = 26C1 x 26C1 x 10C1 x 32C1 x 10C1= 26 x 26 x 10 x 32 x 10 = 2163200 ways.

If it were from A, B, C, D, D: number of ways of choosing passwords = 26C1 x 26C1 x 10C1 x 32C1 x 32C1= 26 x 26 x 10 x 32 x 32 = 6922240 ways.

It is very clear by the above calculations, by comparing the number ways of creating passwords, that choosing one character from each of the Sets, a combination of sets A, B, C, D, D gives 6922240 ways (a very large number in the strings of length 5).

- For strings of length 6:

Here, four characters can be selected from each of one set and the remaining two characters can be from either A, A or A, B or A, C or A, D or B, B or B, C or B, D or C, C or C, D or D, D. Thus, the possible combinations are ABCDAA, ABCDAB, ABCDAC, ABCDAD, ABCDBB, ABCDBC, ABCDBD, ABCDCC, ABCDCD, ABCDDD.

If it were from Sets ABCDAA: number ways of selecting = 146232320 ways.
If it were from Sets ABCDAB: number ways of selecting = 146232320 ways.
If it were from Sets ABCDAC: number ways of selecting = 56243200 ways.
If it were from Sets ABCDAD: number ways of selecting = 179978240 ways.
If it were from Sets ABCDBB: number ways of selecting = 146232320 ways.
If it were from Sets ABCDBC: number ways of selecting = 56243200 ways.
If it were from Sets ABCDBD: number ways of selecting = 179978240 ways.
If it were from Sets ABCDCC: number ways of selecting = 21632000 ways.
If it were from Sets ABCDCD: number ways of selecting = 69222400 ways.
If it were from Sets ABCDDD: number ways of selecting = 221511680 ways.

It is very clear by the above calculations, by comparing the number ways of creating passwords, that choosing characters from the last combination A, B, C, D, D. D gives a greater number of passwords i.e., 221511680 ways among all strings of length 6.

- In the similar way password strings of lengths 7, 8, 9, and so on can be created. From the above observations it is evident that more stronger passwords can be created on the basis of their increasing lengths and using at least one character from the said Sets A, B, C, and D. Once all the above observations are followed then, more and more protection can be obtained from Brute Force and Rainbow Table attacks. All Hackers' attempts will be in vain if all users properly follow the above suggestions properly. Once password protection is obtained then the entire computer networks will be in a safe mode and at the same time information security will also be protected. Rainbow Table attacks are more dangerous comparing to all other kinds of attacks. Surely, users will get protection from various kinds of malicious attacks, which the fundamental secured step in the journey of cyber security in order to provide protection to computer networks. The same secured measures are to be observed from every device connected to the entire networks. Now-a-days for password creations many organisations make restrictions that the password length should be a minimum of 8 characters with a maximum of 15 characters including at least one character from each of the uppercase letters, lowercase letters, digits and special characters. There is a message displaying whether the password is weak, or strong, or stronger. It is to make the stronger in case of its weakness.

II. Conclusion

Analyzing the working mechanism of the Hash table of RBT, all hash functional values are generated by hash functions which are based upon Congruence relations of modular arithmetic. They have high speed cracking ability of passwords as long as passwords are of weak nature. If the nature of passwords becomes stronger their speed will be hindered by the complex organic nature of passwords. Thus, in order to provide a maximum protection from RBT and Brute Force attacks, all users have to strictly adhere to the above suggested rules. More complicated password texts are advisable to create from users' side. Information security has to provide an overall protection to its entire data. Computer network security has to provide security to its systems. These

two conceptual securities will be protected by arresting and minimizing Hakers threats from RBT and Brute Force attacks. It is one of the major steps that users have to follow. There are some other measures that have to be followed.

III. **References**

[1] IEEE Communications Surveys & Tutorials | 2023
[2] Journal of King Saud University - Computer and Information Sciences | 2023
[3] IEEE Transactions on Information Forensics and Security | 2023
[4] Computers & Security - Elsevier | 2023
[5] Journal of Information Security and Applications - Elsevier | 2023
[6] International Journal of Information Security - Springer | 2023
[7] Journal of Cybersecurity - Oxford University Press | 2023