Journal of Data Acquisition and Processing
ISSN: 1004-9037
https://sjcjycl.cn/
DOI: 10.5281/zenodo.7765809

IMPLEMENTATION AND ANALYSIS OF SECURED HASH FUNCTIONS FOR PASSWORD PROTECTION

A.Rajeshkanna
Assistant Professor, Department of Computer Science,Sri S.Ramasamy Naidu Memorial College, Sattur

ABSTRACT

Password protection is the significant process in popular websites as the leakage of password data is increased in numbers. Cryptographic techniques are used for securing the messages which has encryption and decryption process. Generating codes using cryptographic hash functions and algorithms with private and public keys will be used for secured password encryption. In the proposed work, password datasets are encrypted using MD5, SHA-256 and SHA-512 algorithms and the performance analysis of the algorithms in terms of execution time for various password datasets are analysed. If the password datasets are protected using effective fast computation hash algorithms, the attackers cannot guess the characteristics of the data and the password data can be protected with hash values.

Keywords : Hash Function, Collision Resistance, Encryption Datasets, Passwords

INTRODUCTION

Password protection is the primary goal and authentication scheme across the globe in all the websites. Security related to the credentials are generalized to decrease the probability of hacking the password data. Billions of email addresses and passwords are greatly researched and recommendations were suggested for addressing the risks in password protection. Yet the leakage of password datasets itself reveals that the password data among the web surfers and even the highly valuable deemed accounts passwords are also susceptible to be exposed. Many effective algorithms failed in case of cracking with credentials given by the user itself. Most of the secured measures such as strengthening the password with multi factor constraints, minimum password length with alphabets and other symbols and identifying the strength of the password are adopted for authentication and protection. Some of the additional methodologies such as duration policies and expiration of passwords and number of attempts lock out also increases the complexity and helps in protection. This paper follows the following organization: First, the background and related works are discussed. In the proposed work, the algorithms were explained and the methodology of the algorithms were given. The performance analysis of the outputs were discussed with related to the hash functions with different input datasets and concluded.

RELATED WORKS

Cost Asymmetric Secure Hash (CASH) method has been proposed in [1] to minimize the cracking. This was a key based mechanism which was optimized to increase the performance of the algorithm. For the implementation of the algorithm, two datasets from authenticated servers were used. The authors in the reference paper[2], proposed a one-way function to make the password harder to guess. In [3], the authors have introduced Stackelberg game and optimized the parameters of D-hash. In this work, they used the frequently used password datasets to analyse the effectiveness of the hash cost function. The authors quoted in reference paper[4], proposed a method using artificial neural networks and analysed that neural networks can guess passwords easily than other models. The work was implemented with JavaScript with client server model. Jyoti Patil Devaji et.al in [5], implemented SHA-2 algorithm on cryptographic processor for efficient performance of cryptosystem using Verilog hardware description language (HDL).The analysis was performed to find the efficiency of the algorithm. In the related work [6], it was stated that the text authentication can be performed using hash algorithms and the henon map and it was implemented.MD5 algorithm has been modified and randomization tests were carried out for experimental analysis. In the related work [7], an algorithm named SXR (Split, Exclusive OR and Replace) was proposed with the general hash function for securing the password. The authors in the work referenced in [8], an enhanced SHA-1 algorithm was proposed with 512 hashed value answerable for the occurrence of the collisions. Motivated by all these works and the other applications of hash functions in various security mechanisms such as mobile adhoc networks [9], this work has been proposed.

PROPOSED WORK

Cryptographic hash function is used to encrypt credentials such as password and the output of the algorithm will be hash values or it is represented as enciphered text. The purpose of the hash functions may prevent the attackers from hacking and also slow down the process of attackers deciphering. There are different types of hash functions such as salted hashes, keyed hashes and adaptive hashes. The hash functions has certain significant features which was used in password storage and protection. They are Non-reversibility, or one-way function, Diffusion, or avalanche effect, Determinism, Collision resistance and Non-predictable. The proposed work was implemented with three popular hash functions such as MD5,SHA-256 and SHA-512.The input datasets was taken from Rockby dataset and Kaggle datasets.

The methodology of the proposed work flow is as follows:

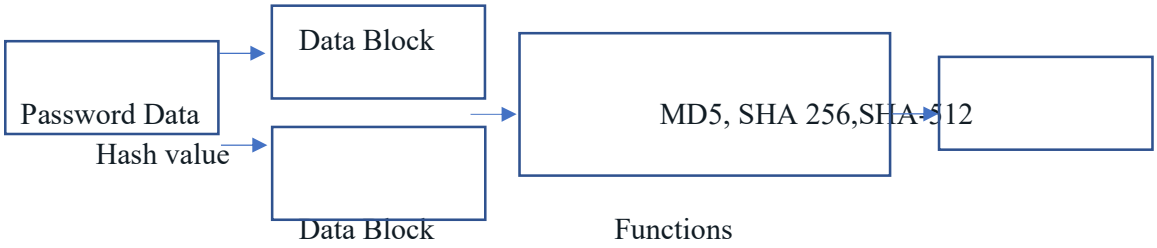

Fig 1. Proposed work

MD5 Algorithm:

The working of MD5 algorithm is as follows:

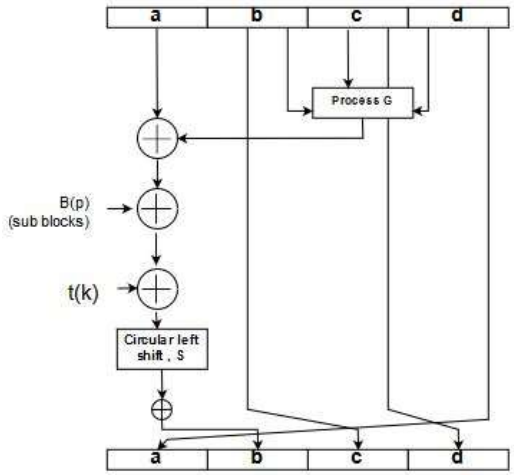

Fig 2. Working of MD5 algorithm

MD5 algorithm has input blocks of 512 bits and padding bits are appended in the message. Single bit 1 followed by 0's are appended to bring the length of the message. In the above figure, the input is divided into a,b,c,d of 32 bits. The data is divided into sub blocks. The processing G will undergo the modulo operations and circular left shift operations. The algorithms were implemented with python-spyder5.1.5 environment. The execution time of the algorithm is calculated for the entire datasets and the sample result of the last four password data in the dataset-1 was given as:

S.No	Digest Message	Time(ms)
1	335764680d9bd9339aa696cdfe25ab0a	75.34106492996216
2	6ec6f024af405c0bc7f4392143445095	75.34106492996216
3	db0b69d1669d2ff15e471f0e99a790b0	75.34106492996216

4	8b77f1a731e5fe9a2668f728cc6ad416	75.34306597709656

Table 1. Output of MD5 algorithm

SHA-256 ALGORITHM:

SHA-256 operations on 512 message block with 64 rounds. First 1 bit is appended and then 'n' bits are appended according to the length L of the bits. The initial message will be exactly 64 bits. In our work, each block has 1024 bits as shown in the below diagram. The working of the algorithm is as follows:

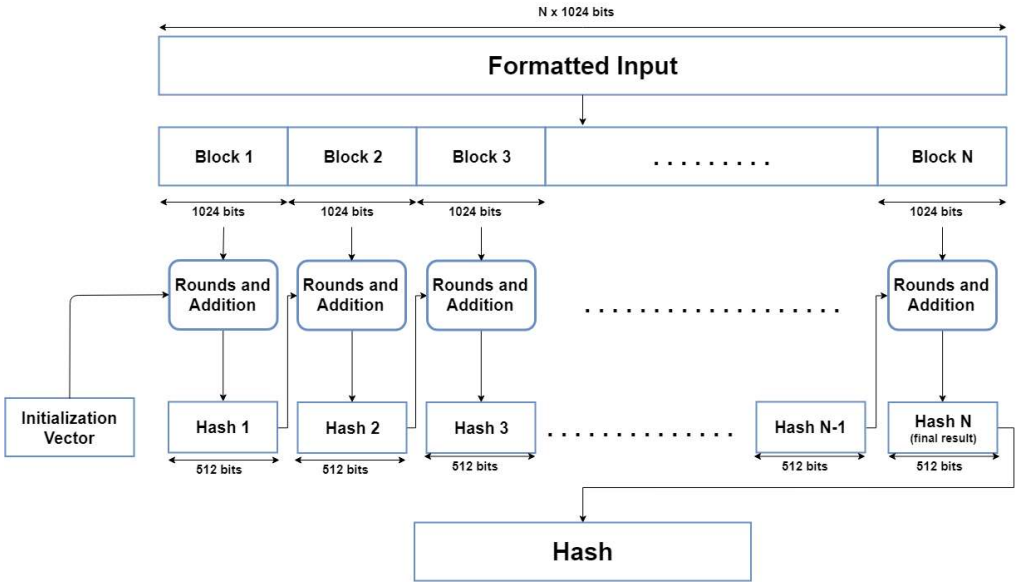

Fig 3. Working of SHA-256 algorithm

The execution time for the last four password data in the dataset-1 is:

S.No	Digest Message	Time(ms)
1	ec5b411b6cf7f06f1d95df68730eb13b1735ec9060047fe784e6655c1d22471a	40.9831321
2	b90b8f700bb4d3152db85bd15b15ec3995466354a1bca3f910297d7f4009a5f1	40.9869742
3	4d9d45a76907f0a05ea606f40e10a7db9feba30c92b636af1562d0028039d2e7	40.9869742
4	4e5a2ba0ac6e6eae21b2cb1e550e8afd6eca8db8897bc03df656f1974629241c	40.9879742

Table 2. Output of SHA-256 algorithm

Working of SHA-512 algorithm:

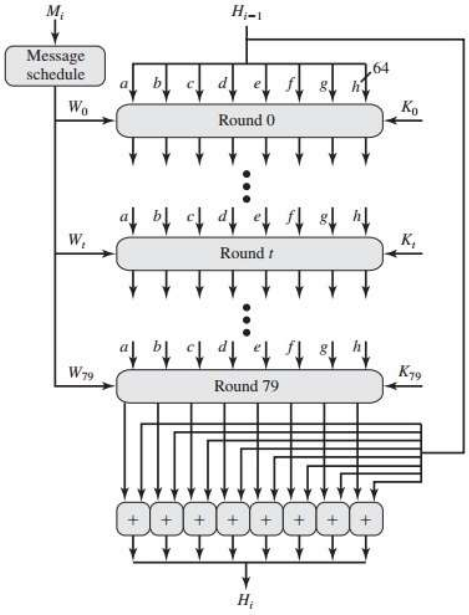

Fig 4. Working of SHA-512 algorithm

The execution time for the last four data in the dataset-1 is:

S. No	Digest Message	Time(ms)
1	a8b60e248fc5d2c64c525aa24952be82e08b9d81e1df63a1038f433b394d19b9e1988391c0894b36fe7e660a8008daee770e2252bf817e70a2d672f474719218	41.5815024
2	5fd800aae7cfd8cf5c25c4deaccb5e1dee78db3dda9c82bc67129d264a5e09f4d7f5ee67a88d754ca90d782ef830f9e7fcc8493595f2b804fa993a27e34f9481	41.5835299
3	77e2cc13369b84b7250b7585120c2e0a2c23d9d33f26522d0348f5b8ab6b708f0ea913ca8737f9b40adbe122f7170cc755c3f9f0f5f3afcd86d17ac7107618f8	41.5835299
4	400c606a50331c3f44df13065b8d205b3857e92bcf80ba9b404d93f01c6ac6c65c351e077d485012874d7180ecaec982dcddaa4a593af9ac6cabc797c6071b0b	41.5835299

Table 3. Output of SHA-512 algorithm

The output of the three types of the algorithms and their execution time for the dataset1 reveals that the MD5 algorithm takes more execution time when compared with SHA-256 and SHA-512. It was tested with dataset2 and the execution time varies in the same way as that of dataset-

1.Further it was found that the digest message value obtained for SHA-512 has performed many operations when compared with SHA-256 and MD5.

CONCLUSIONS

Cryptographic functions are used by focussing on the password protection and other credentials. The popular cryptographic functions are chosen in this work for password datasets. Since the password data will be enormous in numbers, the fast computation time of an algorithm will be useful to prevent hacking attacks, this work has been implemented. The effective hash value was obtained using SHA-512 algorithm and the fastest execution time was obtained with SHA-256 algorithm. This research work will be further high security environments in IOT and other embedded hardware systems.

REFERENCES

1. Blocki, J., & Datta, A. (2016, June). CASH: A cost asymmetric secure hash algorithm for optimal password protection. In 2016 IEEE 29th Computer Security Foundations Symposium (CSF) (pp. 371-386). IEEE.
2. Manber, U. (1996). A simple scheme to make passwords based on one-way functions much harder to crack. Computers & Security, 15(2), 171-176.
3. Bai, W., & Blocki, J. (2021, March). DAHash: distribution aware tuning of password hashing costs. In International Conference on Financial Cryptography and Data Security (pp. 382-405). Springer, Berlin, Heidelberg.
4. Melicher, W., Ur, B., Segreti, S. M., Komanduri, S., Bauer, L., Christin, N., & Cranor, L. F. (2016). Fast, lean, and accurate: Modeling password guessability using neural networks. In 25th USENIX Security Symposium (USENIX Security 16) (pp. 175-191).
5. Devaji, J. P., Iyer, N. C., & Mattimani, R. (2022). Performance Analysis of Secure Hash Algorithm-2 (SHA-) and Implementing on FPGA. In ICT with Intelligent Applications (pp. 1-8). Springer, Singapore.
6. Obaida, T. H., Salman, H. A., & Zugair, H. N. (2022). Improve MD5 Hash Function For Document Authentication. Webology (ISSN: 1735-188X), 19(1).
7. Polpong, J., & Wuttidittachotti, P. (2020). Authentication and password storing improvement using SXR algorithm with a hash function. International Journal of Electrical and Computer Engineering (IJECE), 10(6), 6582-6591.
8. De Guzman, F. E., Gerardo, B. D., & Medina, R. P. (2019). Enhanced secure hash algorithm-512 based on quadratic function. In 2018 IEEE 10th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment and Management (HNICEM) (pp. 1-6). IEEE.
9. Preetha, V., & Chitra, K. (2018). ZBMRP: Zone Based MANET Routing Protocol with Genetic Algorithm and Security Enhancement Using Neural Network Learning. Int. J. Netw. Secur., 20(6), 1115-1124.