Enhancement of MD5 Algorithm for Secured Web Development

Asmin Bhandari[1], Moshiur Bhuiyan[2*], P. W. C. Prasad[1]

[1]School of Computing and Mathematics, Charles Sturt University, Sydney, NSW 2010, Australia
[2] Service Consulting, Enterprise Cloud Systems Pty Limited, Sydney, NSW 2560, Australia

Manuscript submitted January 11, 2017; accepted March 30, 2017.
* Corresponding author. Email: moshiurb@ecloudsys.com
doi: 10.17706/jsw.12.4.240-252

Abstract: The research aims to develop an enhanced MD5 algorithm for the secure web development. The existing MD5 algorithm has many weaknesses that makes it vulnerable to different attacks such as brute force, rainbow table, birthday, dictionary etc. Despite these, the MD5 algorithm is still used in different applications, security protocols and even in the transmission and storage of digital data for verification, integrity and security of data by using checksum. The research focuses on mitigating the weaknesses inherent in the existing MD5 algorithm to secure web application and maintain data integrity and security. The research documents the proposal and implementation of an enhanced MD5 algorithm by varying its length and using a key to hash the data into its cipher form.

Key words: Cryptographic functions, MD5, hash function, rainbow table, brute force, dictionary attack.

1. Introduction

The applications that are accessed by using web browser over a network and developed using web language such as HTML5, CSS3, JavaScript is called as web application [1]. Web application is needed over business to business interaction over network. Web applications has experienced a significance growth over years [2]. The web application has transformed the technology and information sharing platform as it is used in various applications. It has brought a paradigm shift in the way business are developed, delivered, consumed and distributed [3]. However, web applications face a lot of security issues as hijackers continuously tend to attack the vulnerabilities of the web application [4].

Cryptographic hash function is a function that is used to convert any arbitrary length data into hashed output. The data of arbitrary length is considered as an input and hashed output is considered as the 'fingerprint' or 'message digest' of the input. Hashing is a one-way process that is used to convert plain text into cipher text for data integrity, security and protection. The hashed output can never be converted into its original output. Hence, the hashing function is said to be one-way process as contrary to the encryption where data can be retrieved using the key. Data can't be retrieved back in the hashing function. Generally, in web application password are hashed to its cipher form and then matched with stored cipher text to maintain data authenticity. It can be simply termed as a function that converts plain text into cipher text for data integrity, authentication and security [5].

The internet has transformed the world into a global village where people connect and share

different information and data on several platforms. The internet has been a rich mine of valuable information and data which can be extracted not only for better development of society and mankind but also for destruction of society. These valuable information and data can be securely stored by the use of cryptographic hash function along with the protection of user privacy, elimination of data forgery and enhancement of data security. One of the most common cryptographic hash function is MD5 but it has a lot of shortcomings. It can be eliminated by enhancing the algorithm by variable output length and key value pair. Web application can be secured and user privacy can be maintained by the use of enhanced MD5 algorithm to store data securely and maintain its integrity.

The MD5 which stands for Message Digest 5, is a hashing function that was developed by Dr. Ronald Rivest in 1992. Generally, valuable information such as password, valuable data, etc. are converted into hashed form before storage to maintain data integrity and security [6]. MD5 is used as a hashing function in different sector such as web application, security and transmission control protocol, etc. The MD5 function is used in transmission protocol to verify data as MD5 is used to verify checksum. The MD5 hash function can be used to enhance security, efficiency and prevent hacking and deception in many web applications such as e-commerce, content management system. The MD5 algorithm can also be used to prevent data forgery and maintain data integrity too in storage and data transmission [7]. It includes five steps through which the plain text can be converted into cipher text for data integrity and security [8].

2. Background

2.1. Existing MD5 Algorithm

The main MD5 process consist of five steps that is used to convert plain text into cipher text which are as follows:

Step 1: Append padding bits

The message is padded so that its length is congruent to 448, modulo 512. The message is extended so that it is just 64-bit shy of being a multiple of 512 bits long. So, “1” bit is appended to the message and then 0 is appended so that the length is congruent to 448.

Step 2: Append length

A 64-bit representation of length of message before padding bits were added is appended to the result of the previous step.

Step 3: Initialize MD buffer

A four-word buffer is used to compute the message digest where each of the 32-bit register is initialized in hexadecimal, low-order bytes.

Step 4: Process message in 16-word bits

The four auxiliary function is then processed with various steps to produce the desired output.

Step 5: Output

The message digest is produced as an output. The plain text is converted into cipher text or hashed form [9].

The detailed version of the MD5 algorithm can be taken from RFC1321 which is developed by Dr. Ronald Rivest in 1992 [10].

The figure below shows a block diagram of existing MD5 diagram:

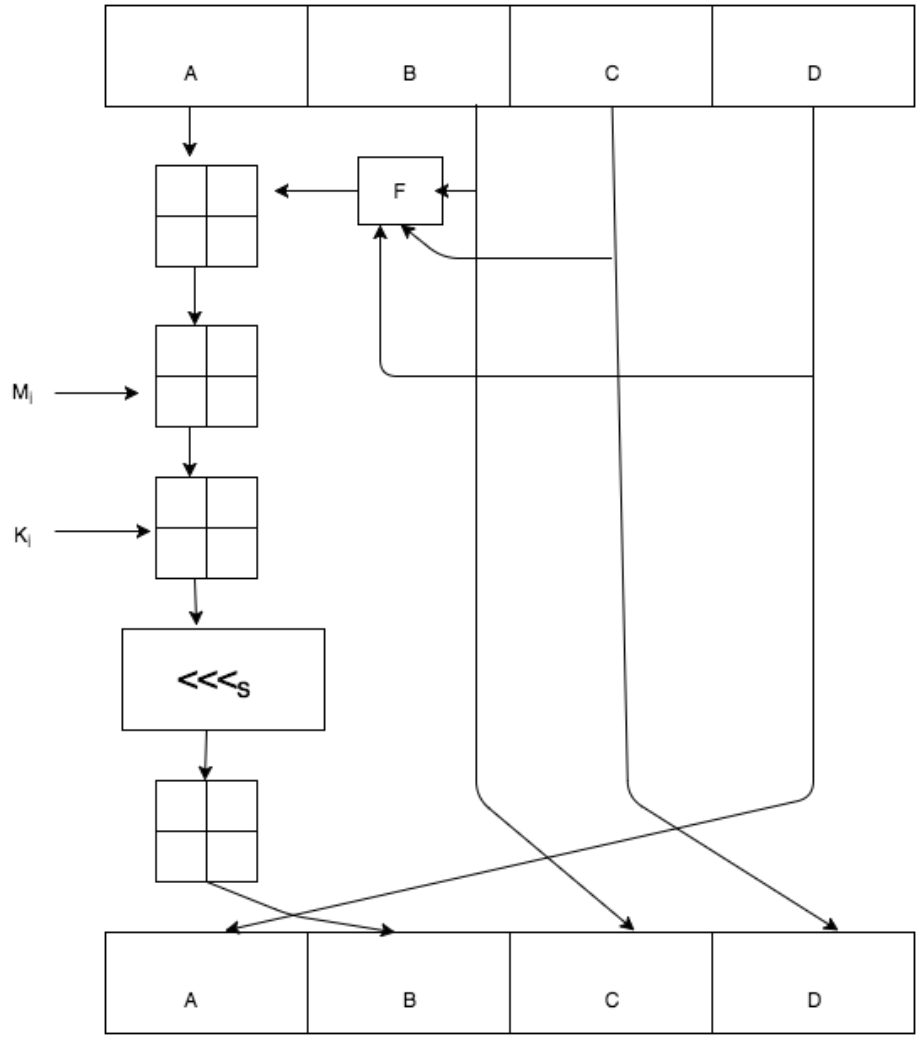

Fig. 1. Existing MD5 algorithm [10].

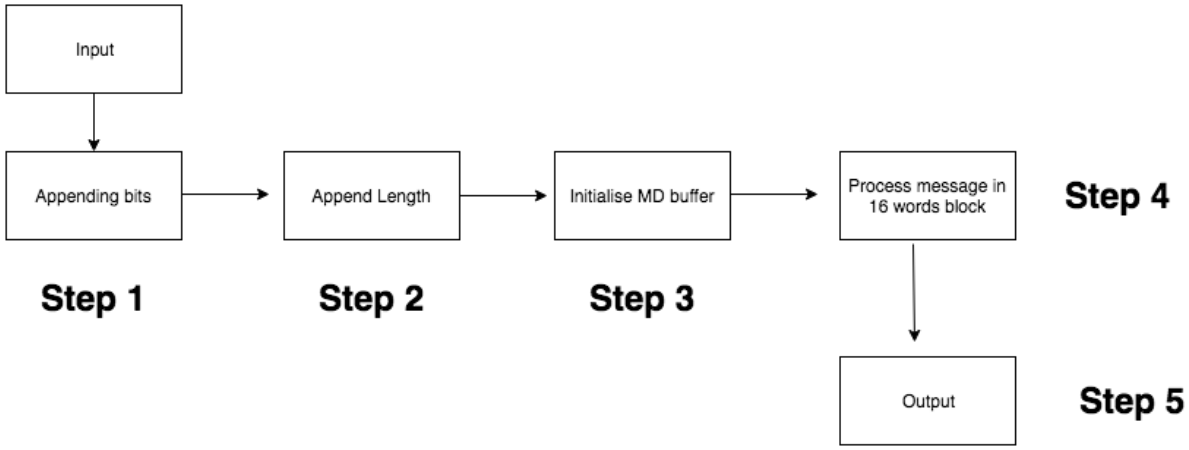

Fig. 2. Block diagram of existing MD5 algorithm。

2.2. Research Purpose and Aims

The purpose and aim of this research is to

- Enhance the existing MD5 algorithm by addressing its existing limitations listed within the literature.

- Propose an enhanced MD5 that can be used for better data security, integrity and hence increase the security, reliability, integrity, performance of the system.
- Maintain user privacy and data by successfully hashing the data that cannot be hacked or hijacked easily.

2.3. Previous Work

A lot of research and analysis has been done in the MD5 algorithm to analyse it from different perspectives. Researches has been done to find out the weaknesses exhibited by the existing MD5 algorithm. Researches shows the weaknesses and vulnerabilities exhibited by the existing MD5 algorithm due to different attacks that prevail. A.so, researches shows the solution for the weaknesses and vulnerabilities exhibited by the MD5 algorithm. Despite these, the solution is not enough to fully secure web application using MD5 algorithm to maintain data integrity and security. Research shows that the MD5 has vulnerabilities and weakness to different types of attacks such as brute force, rainbow table, dictionary, etc. [11]. One of the solution to mitigate the weaknesses of MD5 is usage of a mixed encryption algorithm. A mixed encryption algorithm of MD5 and XOR transformation has been proposed by the author to enhance the security of data. The mixed encryption of MD5 and XOR transformation creates a reliable safety barrier for the system [12]. One of the main weakness of MD5 algorithm is its length where the 128-bits of MD5 is quite small. Since, the length is quite small, a crack of the algorithm is found quite easily. As the length of the data is quite small, the probability to crack the original data is quite high. The weakness related to the MD5 about the length can be easily solved by varying the length of the MD5 algorithm. Researcher suggest that higher level of security and flexibility can be maintained by varying the length of the MD5 algorithm. Also, researcher suggest that the efficiency and security of MD5 with variable output length is better than the existing MD5 algorithm.

Research also suggest usage of MD5 in various data sharing and gathering technique for the effective and efficient operation of the system. A lot of secure communication system and protocols uses hashing methods such as MD5 to maintain communication integrity and signature authentication. MD5 is used as a checksum to verify files and transmission data to verify data integrity. Database designers, security and risk professionals tend to use MD5 to protect data that are vital to the organization in order to maintain data security and integrity. Database integrity and security are also vital in database development where hashing function such as MD5 is used to convert plain text to cipher text before storage in database. Research also suggests the use of salt technique to eliminate the threat possessed by pre-computed rainbow table attack and dictionary attack. The research proposes usage of a key to eliminate the threats exhibited by the rainbow table attack and dictionary attack [13]. Research also suggest the usage of a hybrid algorithm using MD5 and RSA encryption algorithm to maintain data integrity and verification in networks such as mobile ad-hoc network. It focuses on a secure data transmission network such as mobile ad-hoc network [14]. Research also suggest the usage of MD5 in digital signature, key exchange protocol where the MD5 technique is used to verify data in public key cryptosystem and compression of file to check and verify files on the other end.

Furthermore, researches shows the usage of MD5 to propose a new cryptosystem where the data is hashed by using MD5 technique to verify data by using unused bits in TCP header for the maintenance and implementation of data security, integrity, verification and authentication. The research also shows the usage of a key and MD5 algorithm to eliminate rainbow table and dictionary attack. The

crack technique such as rainbow, dictionary and birthday can be used to crack MD5 algorithm quite quickly as the hashed output is already cracked. Such situation can be easily mitigated by using key technique. In addition to data transmission and authentication, digital watermarking can also be obtained by using MD5 where block based watermark is embedded to prevent illegal copying and duplication of digital media. In addition to it, research also suggests that cryptographic functions such as MD5 helps to maintain the confidentiality, integrity and authenticity of digital data such as digital images over open network that are often considered insecure because of malicious activities. Usually a secret shared key is generated between the terminals communicating with each other and then hashed using a MD5 functions [15].

Similarly, research suggests that the cryptographic algorithms are quite relevant in the communication and storage of digital data to secure data. Hence, a lot of architecture also has been proposed to implement MD5 algorithm in the processor which makes the use of MD5 easy and relevant for data security. MD5 is used to verify data by using MD5 checksum to maintain data integrity and security in transmission. Also, research shows that the cryptographic hashing function can be made secure by usage of key with the cryptographic function between sender and receiver.

According to Merkle-Damgard construction, if the basic function that sum up to build a cryptographic function is collision resistant, then the cryptographic function is said to be collision resistant. According to Merkle-Damgard construction, if the compression function h is collision resistant in the above diagram, then the cryptographic function H is said to be collision resistant [16]. Since, the compression function that is used to create the cryptographic function is collision resistant, the cryptographic function will be collision resistant [17].

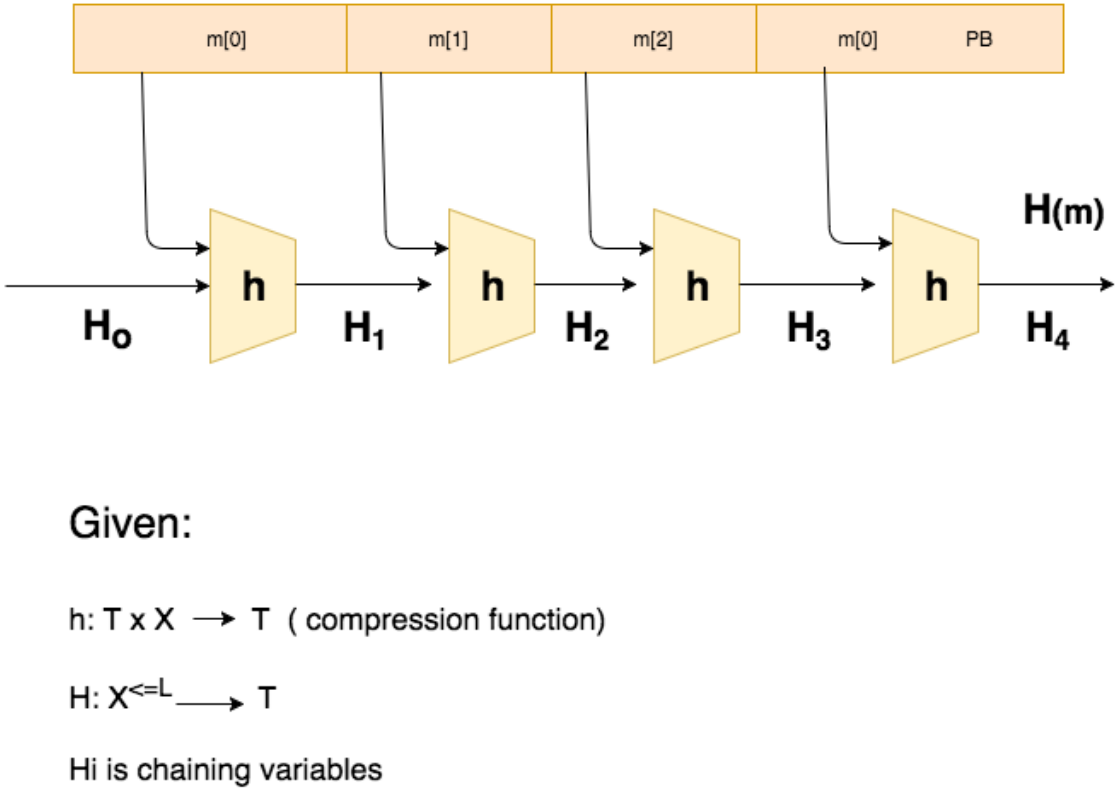

Fig. 3. Merkle-Damgard construction.

3. Weakness in the Existing MD5 Algorithm

The MD5 algorithm exhibits a lot of weaknesses such as its vulnerabilities to different attacks such as rainbow table, dictionary, birthday, etc. A lot of research has been performed to find out the weaknesses of the MD5 algorithm. The literature review focuses on the different attacks that can be performed on the MD5 algorithm. A lot of paper focuses on the different type of attacks that can be performed on MD5 to crack or hijack the algorithm.

Brute force attack

Brute force is an exhaustive search technique that can be used in any of the data encryption or hashing function. It finds the original input or message by using every possible combination of data to find the original message. In scenario to password cracking, it uses all possible combination of characters to find out the original password. The brute force attack is quite common technique where the original data is cracked by using every possible combination and permutation. The brute force attack is a crack technique that can be used against any encryption data where all possible number of combinations and permutation can be applied to find out the original message.

Birthday attack

A birthday attack is an attack where the attack works by effect of chance where the process is random and more as a guess. The attack works by the roughly random process and probability as in the birthday paradox. It threatens the message integrity and data security. The birthday paradox is a scenario where the probability of two people having birthday on the same date is quite high if the number of people in the room is quite high. The birthday paradox is used to exploit the weakness of the cryptographic functions [18]. It is quite different to the brute force scenario where every possible data is tested until the hashed output are identical. Birthday attack is quite faster and efficient than brute force attack. The length of the data is quite significant in the birthday attack. Birthday attack is more likely to be successful if the hashed output is quite small in length [19].

Dictionary attack

A dictionary attack is an attack where the original message is found out by using list of dictionary data that are usually used as password. It contrasts to brute force attack where all of the possibilities are used to hijack the cryptographic function. Contrary to the brute force, it uses a guess to hijack the algorithm by using list of possibilities. Since, the attack uses a guess, it is more likely to hijack as compared to that of brute force. However, the hijack speed of dictionary attack is quite faster than that of brute force technique. Dictionary attack precomputes the hashed data of the dictionary files. The generated hashed data is matched against the data to be cracked or hijacked [20]. Rainbow table is a more refined version of dictionary attack. Generally, a small password along with small domain enables the hijacking of data quite easy using dictionary attack [21].

Rainbow table

A rainbow table attack is an attack where the original message is found out by using a table that exist by computing the hashed output for various input by using the same functions. The rainbow table is quite faster as the output is already stored for faster hijacking of the algorithm. The hashed output for the plain text is pre-calculated and stored in the database for faster and efficient hijacking of the cryptographic algorithm such as MD5. A rainbow table is an enhancement of brute force technique where a rainbow table is actually a precomputed table consisting of hashed value that consist of original message and its hashed output. Also, rainbow table is also a refined version of dictionary attack where the hashed output are data that have more probability to hijack the data. Since, the

hashed output for a same original data is same for all cases, the time to compute the desired hashed output is drastically reduced. So, it is one of the major weakness for the MD5 algorithm as the hashed output for the same input is always same.

4. Proposed Enhancement to MD5

A lot of research papers and journals have been conducted in order to devise a solution for the weaknesses exhibited by the existing MD5 algorithm. The literature review shows all the advantages and disadvantages of each process used in the existing MD5 algorithm. So, the weaknesses have been discarded and strength or advantages has been considered. One of the major weaknesses of MD5 algorithm is that the hashed output for the same input value or data is always same. This technique can be mitigated by using a key value that differentiates the output for the same input. Also, brute force attack shows that the algorithm can be easily hacked if the length of the hashed output is small. Hence, this weakness can be mitigated by simply extending the length of MD5 algorithm. So, the solution to the weaknesses of the MD5 algorithm is usage of key and variable output length. So, the research proposes and documents the implementation of hybrid model of both the key value and variable length that is a solution to weaknesses exhibited by the existing MD5 algorithm.

4.1. Enhanced MD5

Since, existing MD5 algorithm has a lot of weaknesses and shortcomings, it can be mitigated by using an enhanced version using a mixed algorithm of variable output length and key value. The enhanced MD5 algorithm enhances the existing MD5 algorithm by using key and varying its length. So, the enhanced algorithm uses a mixed algorithm of variable output length and key. Since, the key is different for each user, the output varies for different key but however same key yields same output. The use of key in enhancement of MD5 algorithm completely eliminates the threat of rainbow table, birthday and dictionary attack as the output will be different for different key. Since, the length of hashing function makes it quite vulnerable to different attacks, variable length will enhance the hashing function. The figure shows an enhanced version of existing MD5 algorithm.

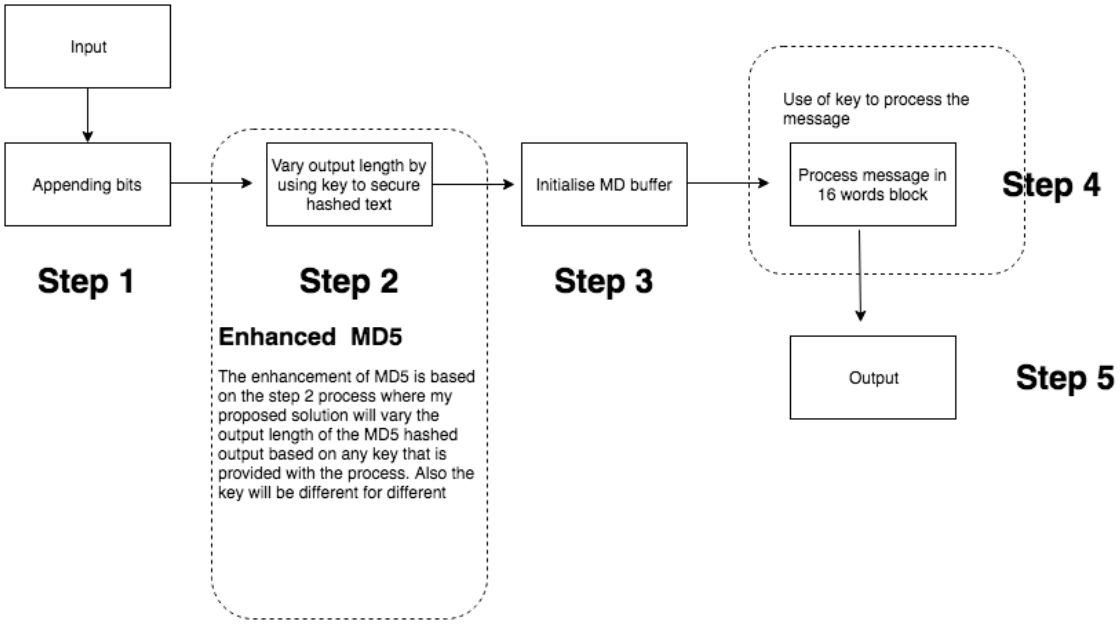

Fig. 4. Block diagram of enhanced MD5 algorithm.

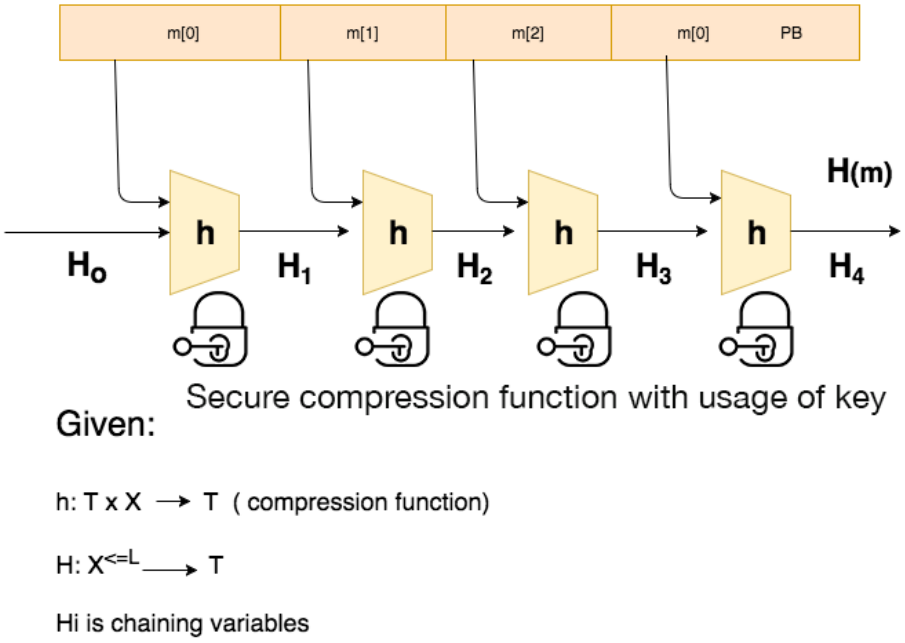

Fig. 5. Enhanced merkle-damgard construction for enhanced MD5.

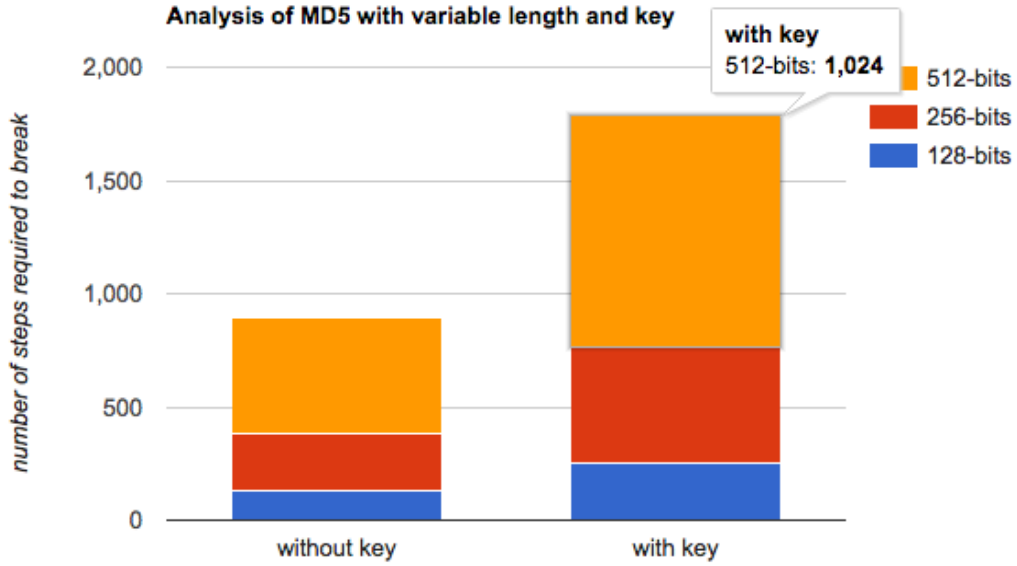

Fig. 6. graph showing analysis of enhanced MD5.

Bits	Permutation	Rounds	Operation required to break
128	2^128	4	2^128
256	2^256	8	2^256
512	2^512	16	2^512
1024	2^1024	32	2^1024

Fig. 7. Table showing bits, permutation, rounds and operation required to break.

5. Web Implementation

The weaknesses and shortcomings of MD5 can be mitigated by using a hybrid algorithm of key-value pair and varying length. The existing MD5 algorithm produces a 128-bit output data. So, we can mitigate the shortcomings and weaknesses of the MD5 algorithm by varying the length of the MD5 algorithm. Also, the MD5 algorithm can be further enhanced by producing a hashed output on the basis of a key value. The use of key value pair for enhancement of MD5 completely eliminates the threats from rainbow table, dictionary and birthday attack. Different key is used to produce a hashed output. Different key produces different hashed output but only same key can produce same hashed output.

The hybrid algorithm of key-value pair and varying output length is implemented by using web technologies. The implementation shows the use of enhanced MD5 algorithm in web application for data integrity and security. The web technologies used in this process are PHP, HTML5, CSS3 and JavaScript. PHP server has been used as a backend server for implementation of the enhanced MD5 algorithm. The front end implementation has been done by using web technologies such as HTML5, Javascript and CSS3. The enhanced MD5 algorithm has been implemented by using JavaScript. The images below show the implementation of the proposed enhanced algorithm using web technologies:

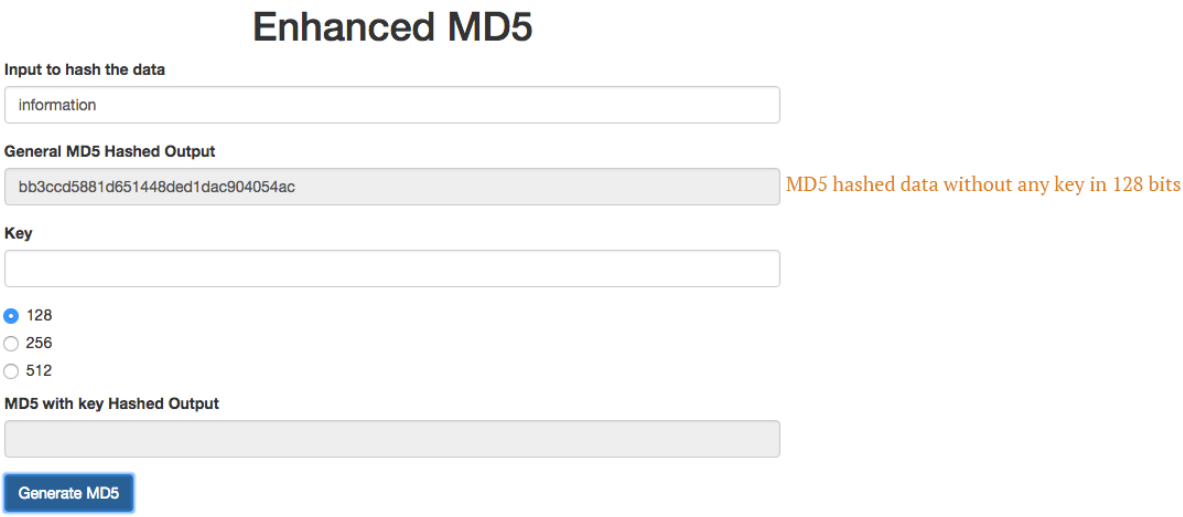

Fig. 8. 128-bit hashed MD5 output.

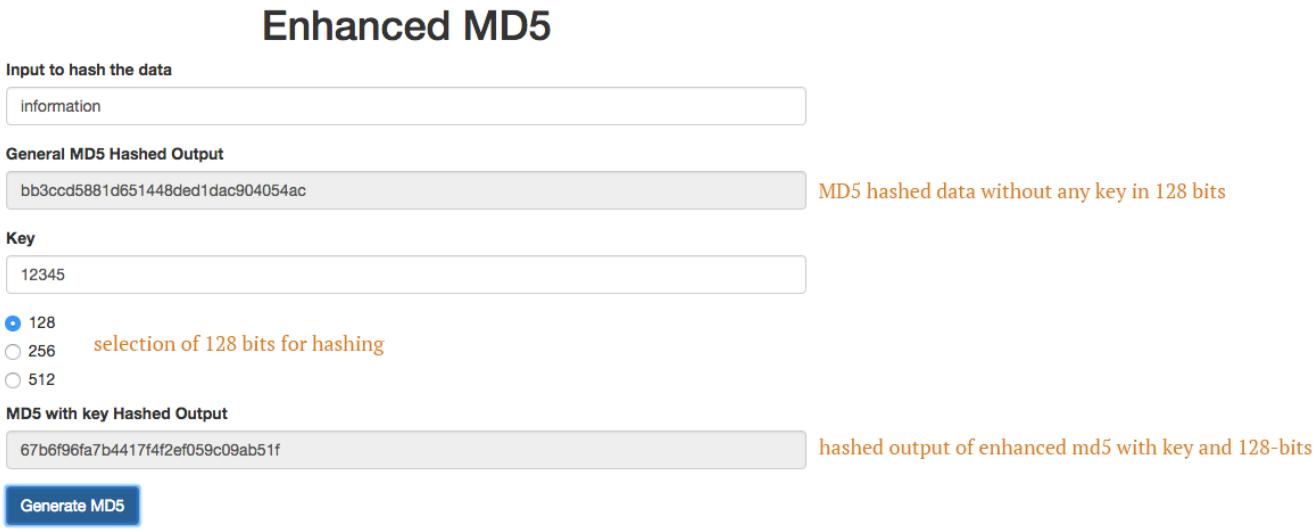

Fig. 9. 128-bit hashed output with key.

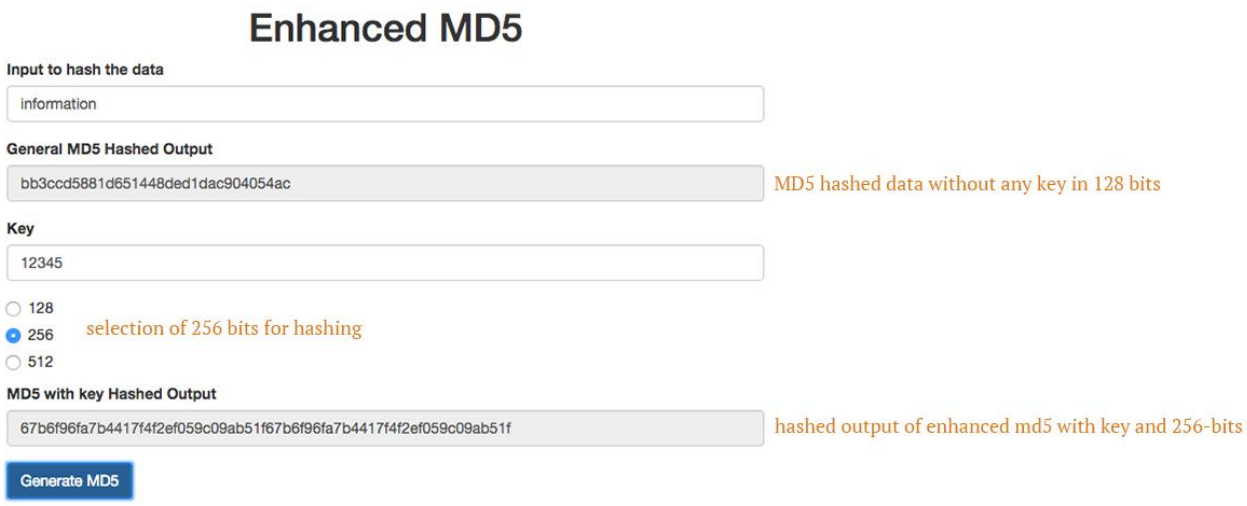

Fig. 10. 256-bit Hashed MD5 output with key.

Fig. 11. 512-bit hashed MD5 output with key.

6. Discussion and Evaluation

The MD5 is a one-way hashing function or technology used in different applications, security protocols and even in the transmission media to maintain data integrity, authenticity, security by converting plain text or data into cipher text. The MD5 hashing function is a one-way hashing function which means that the hashed output cannot be converted back into its original format. In such cases, the plain text is considered as an input which is of arbitrary length and output is the “fingerprint” or “message digest” of the respective input. An example of it can be found in the web technologies where the important data such as password, credit card numbers, etc. is often hashed before storage to maintain data integrity and security. Usually, database designers and security personnel hash the important and valuable data before storage to maintain data integrity, security and flexibility. Also, the MD5 algorithm is used in security and transmission protocols along with digital signature. MD5 is also used to verify data during transmission by using MD5 checksum. Generally, password is hashed into cipher text before storing in the database in web application to maintain data security and integrity. However, the algorithm exhibits a lot of weaknesses and vulnerabilities to attacks such as brute force, rainbow table, dictionary, birthday, etc.

Despite having vulnerabilities to many attacks, the algorithm can be enhanced and implemented in the existing domain where the algorithm is being used. The MD5 is enhanced by using an enhanced algorithm of varying output length and key-value pair. The length of the output of MD5 is varied upon different length such as 256, 512 and more as the length of MD5 is often considered as a weakness to hijack the algorithm. Also, a key is used to convert the plain text and data into cipher form where only the same key can yield the same output. Since, the key is different for each unique user, the MD5 is more enhanced as it reduces the risk of being hacked by using rainbow table as the hashed output will be different for different key. The use of key eliminates the rainbow table attack where the output can never be pre-computed as it varies for different key. Also, the use of key eliminates the threat of brute force attack as the output heavily depends upon the key. One of the major weakness of MD5 is its length where it is often hacked as the length is only 128-bits. So, the research proposes a usage of varying length. Since, the length of the MD5 is increased and varied, it will be enhanced resulting in more secure and safe implementation of the algorithm. The proposed algorithm enhances and refines the existing MD5 algorithm and hence it can be used for safe, reliable and secure implementation. Also, the research documents the implementation of enhanced MD5 algorithm for secure web application to maintain data integrity, authenticity and security.

7. Conclusion

Since, existing MD5 algorithm exhibits a lot of weaknesses and vulnerabilities against different attacks such as birthday, brute force, rainbow table and birthday, it can be easily mitigated by using a hybrid algorithm of variable output length and key technique. The MD5 algorithm is used in various sectors for digital signature, data integrity, security, authenticity along with security and transmission protocols. Therefore, a lot of security issues and threats are generated because of its vulnerabilities. The weaknesses can be easily mitigated by varying the length of the existing MD5 algorithm. Also, it can be further enhanced by using key to avoid attacks such as birthday, brute force, rainbow table and dictionary. Hence, the proposed enhanced algorithm introduces an enhanced version of the MD5 algorithm by using a mixed technique of variable length and key.

The paper documents the proposal and implementation of enhanced MD5 algorithm using web technologies such as JavaScript, HTML5, CSS3 and PHP. The existing MD5 algorithm has a lot of weaknesses and threat to lots of attacks such as brute force, rainbow table, dictionary, etc. The research proposes a hybrid algorithm to enhance the MD5 algorithm using key technique and variable output length. Since, the length of MD5 is only 128 bits, solution of using variable output length increases the security and effectiveness of the MD5 algorithm. The security, integrity and effectiveness can be further enhanced by using a key to hash the plain text or data into cipher form to maintain data security and integrity. The enhanced MD5 is implemented by using web technologies i.e., PHP, JavaScript, HTML5 and CSS3.The server used to implement the enhanced MD5 is PHP server. The enhanced MD5 is implemented using Javascript. Thus, the data integrity, security and flexibility can be maintained by using the proposed mixed algorithm of variable output length and key technique. Hence, we have proposed and documented the implementation of enhanced MD5 algorithm by using variable output length and key technique.

The process of enhanced algorithm can be limited by the speed of data storage and originality of stored data as the original data is never retrieved once the data is processed through cryptographic hash function. Since, the data needs to be processed through cryptographic hash function for the

operation of web application, the execution time of web application will be slower. However, the usage of cryptographic function will maintain data integrity and security.

Appendix: Codes to Implement and Test the MD5 Extension

The codes to implement and test the MD5 extension can be found online and downloaded from the following link:
http://asminbhandari.com/docs/

References

[1] Al-Fedaghi, S. (2011) 'Developing web applications'. *International Journal of Software Engineering and Its Applications*, *5(2)*, 57–68.

[2] Lomte, V. M., Ingle, D. R., & Meshram, B. B. (2012). 'A secure web application: E-tracking system', *International Journal of UbiComp*, *3(4)*, 1–18.

[3] Maan, J. (2012). 'Mobile web — Strategy for enterprise success'. *International Journal on Web Service Computing*, *3(1)*, 45–53.

[4] Rafique, S., Humayun, M., Gul, Z., Abbas, A., & Javed, H. (2015) 'Systematic review of web application security vulnerabilities detection methods. . 28–40.

[5] Ora, P., & Pal, P. R. (2015). Data security and integrity in cloud computing based on RSA partial homomorphic and MD5 cryptography. *Proceedings of the 2015 International Conference on Computer, Communication and Control.*

[6] Wang, M. J., & Li, Y. Z. (2015). Hash function with variable output length. *Proceedings of the 2015 International Conference on Network and Information Systems for Computers.*

[7] Zhenggang, H., & Yueming, L. (2014). A method based on MD5 and time for preventing deception in electronic commerce. *Proceedings of the International Conference on Cyberspace Technology.*

[8] Yu, W., Jianhua, C., & Debiao, H. (2009). A New collision attack on MD5. *Proceedings of the International Conference on Networks Security, Wireless Communications and Trusted Computing.*

[9] Gupta, P., & Kumar, S. (2014). *(IJCSIT) International Journal of Computer Science and Information Technologies*', *A Comparative Analysis of SHA and MD5 Algorithm*, *5(3)*, 4492–4495.

[10] Rivest, R. (1992). MIT laboratory for computer science and RSA data security, Inc. *The MD5 Message-Digest Algorithm', The MD5 Message-Digest Algorithm*, 1–21.

[11] Kasgar, A. K., Dhariwal, M. K., Tantubay, N., & Malviya, H. (2013). I*nternational Journal of Modern Engineering & Management Research'*, *A Review Paper of Message Digest*, *1(4)*, 29–35.

[12] Nan, B. X., & Xiang, D. H. (2010). The mixed encryption algorithm based on MD5 and XOR transformation. *Proceedings of the 2010 Second International Workshop on Education Technology and Computer Science (ETCS).*

[13] Jacob, N. M. (2015). Vulnerability of data security using MD5 function in PHP database design. *EPH International Journal of Science and Engineering*, *11(1)*, 11–15.

[14] Singh, K., & Goel, C. (2014). Using MD5 AND RSA algorithm improve security in MANETs systems. *International Journal of Advances in Science and Technology*, *2(2)*, 48–54.

[15] Zheng, X., & Jin, J. (2012). Research for the application and safety of MD5 algorithm in password authentication. *Proceedings of the 2012 9th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD).*

[16] Backes, M. *et al*., (2012). Verified security of merkle-damgard. *Proceedings of the 2012 IEEE 25th Computer Security Foundations Symposium*.

[17] Joseph, M., Wachiuri, T., & Cheruiyot, W. (2014). Enhanced message digest version 5 architecture for secure hashing. *IOSR Journal of Computer Engineering*, *16(5)*, 125–129.

[18] Kim, L. M. (2015). 'The mathematics behind the birthday attack. *Principia: The Princeton Undergraduate Mathematics Journal*, 20–23.
[19] Gupta, G. (2015). *What is Birthday Attack?*
[20] Majumder, J. (2012). Dictionary attack on MD5 hash. *International Journal of Engineering Research and Applications (IJERA)*.
[21] Pinkas, B., & Sander, T. (2002). Securing passwords against dictionary attacks. *Proceedings of the 9th ACM* conference *on Computer and Communications security*.
[22] Cao, D., & Yang, B. (2010). Design and implementation for MD5-based data integrity checking system. *Proceedings of the 2010 The 2nd IEEE International Conference on Information Management and Engineering*.

Asmin Bhandari was born in 1992 in Kathmandu, Nepal. He completed his masters of information technology with Distinction from Charles Sturt University. Currently, he is working as a software developer and specialises in application development.

Moshiur Bhuiyan is an experienced IT Management Consultant, who possesses extensive expertise in management consulting, business analysis, BPM, change management and enterprise architecture. He has significant passion in research and teaching. His research areas include but are not limited to Business Process Discovery & Modelling, Process Rules and Policy Integration, Process Execution, Process Reengineering and Optimization, Process Lifecycle Management, Change Management, Software Requirement Engineering, Cloud Computing, ICT Governance & Architecture. He has published his works in reputed international conferences and journals. He has served as program committee member and reviewer in several conferences and workshops. He is also the founder member of a technology entrepreneurship company named Enterprise Cloud Systems (www.ecloudsys.com) which develops innovative cloud applications.

P. W. C. Prasad is an adjunct associate professor with the School of Computing and Mathematics at Charles Sturt University, Australia. Prior to this, he was a lecturer at the United Arab Emirates University in UAE, Multimedia University in Malaysia and also the Informatics Institute of Technology (IIT), Sri Lanka. He gained his undergraduate and postgraduate degrees from St Petersburg State Electrotechnical University in the early 90s and completed his PhD studies at the Multimedia University in Malaysia. He is an active researcher in the areas of computer architecture, digital systems, modelling and simulation. He has published more than 100 research articles in computing and engineering journals and conferences proceedings. He has co-authored two books entitled 'Digital Systems Fundamentals' and 'Computer Systems Organization and Architecture' published by Prentice Hall. He is a senior member of the IEEE Computer Society.