

Data Integrity Checking for Securing Students Grades Lists Using the Hash Function

Megah MULYA¹, Hardini NOVIANTI², Des Alwine ZAYANTI³, Syahrul Ramadhan A.S.⁴, Rendy WIJAYA⁵, and Christofer YEREMIA⁶

^{1,2,4,5,6}*Informatics Engineering, Computer Science Faculty, Sriwijaya University, Indonesia*

³*Mathematics, Faculty of Science, Sriwijaya University, Indonesia*

ABSTRACT

Presently, institutions should preserve the quality of teaching, services and resources to ensure the quality of the institutions. All information is presented in full, fast and always in accordance with existing developments. The use of computer technology has an impact on eliciting gaps by irresponsible parties. The list of grades or transcripts of student grades is one of the objects targeted by the irresponsible parties experienced by several institutions in Indonesia. The fact that the list of student values in an institution is very important, a mechanism is needed to secure data manipulation from irresponsible parties. For the purpose of securing the list of grades, cryptography techniques are used. Data integrity, part of authentication in cryptography, is the right choice for this purpose. This study uses the data integrity method by comparing the initial hash value and final value. In this study the integrity of the data is examined using the SHA-512 algorithm which has the one-way function. A scheme and n algorithm which can be used to check changes in values in the database has been created to ensure the integrity of the students grades. Furthermore, a software prototype that has been developed can be used to check the integrity of student grades.

With the prototype software, it is proven that if student data has been changed by an unauthorized party, a warning should appear. Changes in value that have occurred can be reported by using program history which can be used as evaluation material in the future. To provide a warning to the leader of a faculty, a pop-up notification containing a report message is made each time there is a change in the data value without the right permissions. The pop-up notification is generated by a trigger in the database and thus perform a comparison of hash values using the right query in an application.

Keywords: *integrity, SHA-512, list of grades*

INTRODUCTION

In education, especially higher education, there is currently intense competition of information between institutions. All information is presented in full, fast and always in accordance with existing developments. Therefore, the information needed is processed by a computer system that can produce information based on required criteria. On the other hand, the advantages provided by the computer system have a negative impact in the form of a gap deviation. In education, the existence of transcripts of students grades is crucial because it is the final result that illustrates the ability of students. Students transcripts is of great significance to the institution to provide an overview of student abilities. A bias in scoring is highly undesired. This bias may be caused by the quality of education or the crime of manipulating students grades data done by certain parties. Quality of education which causes a bias in assessing the ability of students should seek for solutions to improve the quality of various academic community especially the ability and morals of the lecturers. The crime of buying and selling values at institutions by means of data manipulation may even be done by the administrator. Crimes which is resulted from manipulation of grades by unauthorized parties can have a long-term negative impact

on parents of students, alumni users and institutional credibility. Therefore we need cryptography techniques that may be used to secure confidential data in the form of students grades data is a data-integrity checking technique. For the purpose of guaranteeing integrity, one-way functions are used (Hash functions) with various algorithms in between SHA-512, SHA-256, MD5 and so on. In this following study, the SHA-512 algorithm will be used due to its collision-free properties [8], and compared to SHA-256, this algorithm has a higher speed by 37.5%[7].

METHOD PROPOSED

Hash function

A hash function is a mathematical function that converts numeric input values into compressed numeric values. The input of hash functions may vary in length, but the output of the hash value will always have a fixed length. The hash function is used for the purpose of ensuring integrity. Commonly used hash functions are the MD group and the SHA group algorithms are MD1, MD4, MD5 and SHA-1, SHA-128, SHA-256, SHA-512 respectively because these two groups of algorithms are quite fast and safe. Both of

these algorithms each have their own strengths and weaknesses. Hence, its use is adjusted to the characteristics of the problem at hand. Numerous debate has occurred about which algorithm is safer to check the authenticity of messages because a crack has been found to resolve MD5 and SHA-1 [2].

In case of safety, MD5 hash function algorithm is no longer safe from collision that may happen with this algorithm. In the certain cases, SHA-512 is considered as a secure hash function until this day, because of the possibility of data collision is almost nonexistent[6].

In terms of complexity, a study by Piyuh Gupta has stated that the MD5 and SHA256 algorithms have the same complexity and the value is $O(N)$. But in study done by S.Gueron, SHA-512 is 37.5% faster than SHA-256 due to the fact that SHA-512 has less round for more byte processed than SHA-256[7].

In cases where speed is an important issue, various optimizations need to be carried out to allow periodic checks on stability. In addition, empirical data should be used to determine which algorithm will be applied in a case. Thus, data analysis and extrapolation of empirical results will help in determining which algorithm is most suitable for specific requirements.

Integrity Scheme

The integrity mechanism could be implemented with a cryptography algorithm in investigating the authenticity of the text file that is accessed with the original text file at the beginning of storage by matching the hash values of the two files. Files that are authenticated are valid as messages. Message storage and message access can ensure the integrity and authenticity of messages by utilizing the hash function in cryptography. Figure 1 shows the message integrity testing with the hash function [3].

In Figure 1, before the M message is sent or accessed, it is processed by using the H hash function that generates the old digest message $h = H(M)$ and is stored as a reference. The next time a message is received or accessed by M' , will it be tested whether the message sent or accessed initially is the same as the message received or accessed later. In other words, is $M = M'$? Then, recalculate the new digest message $h' = H(M')$. If $h' = h$ means it can be concluded that the message was not manipulated during the sending process [4].

Since the process of ensuring file integrity can be examined by comparing the hash value of the original file at the beginning of storage with the hash value of the file that you just accessed, if the hash values obtained from the two files are the same, then the file's integrity and authenticity are confirmed, whereas if the hash value obtained from the two files are different then it can be concluded that the received file is not intact or has been manipulated. The implementation of this algorithm can be used to determine the authenticity of a file [5].

Implementing the cryptography system hash function for the purpose of checking file integrity should prevent the manipulation of files that will be accessed in the future, especially in multi-user environments.

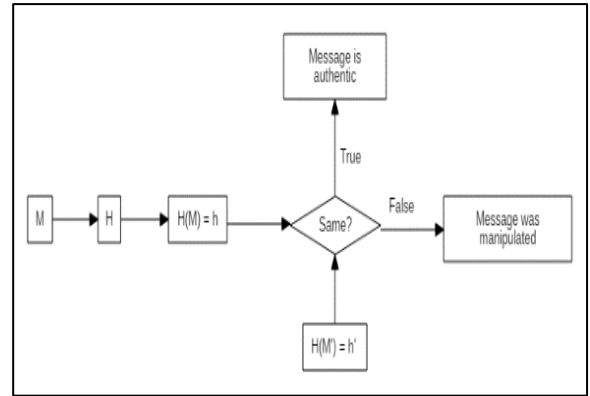


Figure 1. Checking Message Integrity Chart

By combining the principles of hash functions and making integrity schemes, the research has been executed in these following steps :

1. Creating an integrity scheme
2. Designing database
3. Creating test case to guarantee students grades data integrity
4. Testing students grades integrity testing using black box method

RESULTS AND ANALYSIS

Students grades are secured with the SHA-512 algorithm. Students grades data and their hash values are collected in the database. A trigger was created to give a warning to the leader preserved in the database.

Integrity check chart

By using the integrity check chart from Figure 1, the student grades integrity check scheme is created. This scheme involve database, which represents the students grades M and hash value for the students grades $H(M)=h'$. Using Figure 1 as the base for checking student's grades integrity and initiate actions that will be performed when comparing values results in true or false. Complete scheme can be seen in Figure 2.

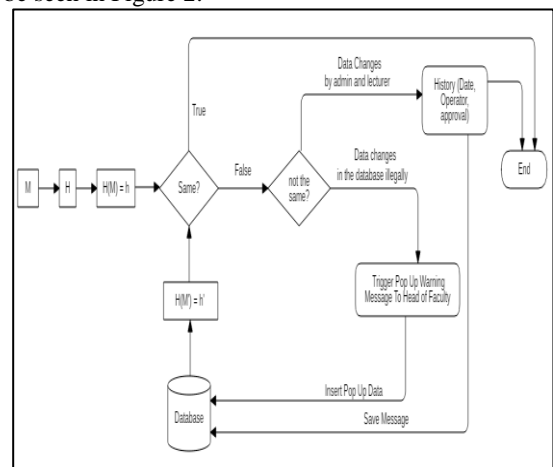


Figure 2. Checking The Integrity of Student Grades Scheme

Trigger Development

In this research, one the implementations of SHA-512 algorithm is this algorithm is used on trigger before

updating students grades database which contains some comparisons to check the integrity of students grades. The algorithm can be seen below:

Table 1. Pseudocode to check students grades integrity BEFORE UPDATE on database

```

Algorithm 1 : BEFORE UPDATE TRIGGER
START
  IF SHA-512( NEW.StudentGrades) !=
  StoredHashValue
    Gives pop-up warning to academic leader;
  END

```

The impact caused by the trigger is that every change in the value in the student_score table where the result of the hash value does not match with the stored hash value, data will be entered into the pop-up table. This will cause a pop-up notification to appear on the leader's account which can be seen in Figure 3.

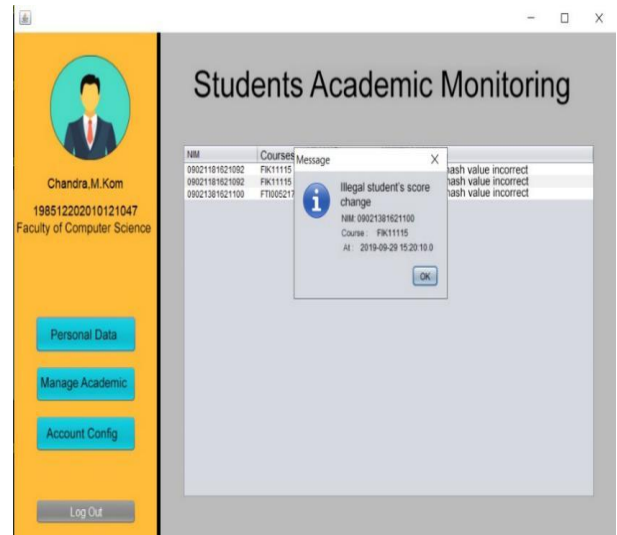


Figure 3. Pop-up view of academic cheating

Management monitoring algorithm

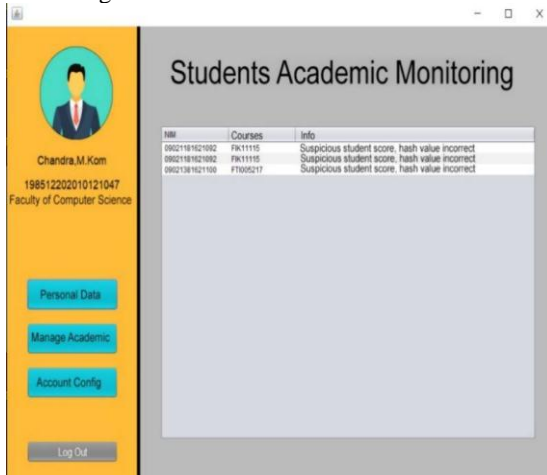
Table 2. Pseudocode of Student's grades comparison

```

Algorithm 2 : Student's Grades Comparison
START
  Array of StudentGrades = GetData FROM Database
  FOR EACH StudentGrades
    START
      String HashData = GetHashForSelectedStudent FROM Database
      IF SHA-512(SelectedStudentGrades) != HashData
        Show student data in table
      ENDIF
    ENDFOR
  END

```

The algorithm above is developed to detect any mismatch between students grades and their hash values. If there is an inconsistency during the comparison, then it will be displayed in the table provided in the monitoring menu, as seen in figure 4.



NW	Courses	Info
0902181621092	FK11115	Suspicious student score: hash value incorrect
0902181621092	FK11115	Suspicious student score: hash value incorrect
0902181621100	FK00027	Suspicious student score: hash value incorrect

Figure 4. Display of academic cheating monitoring records

CONCLUSION

SHA-512 algorithm could be applied to maintain the integrity of student data in the form of grades. Supposing that the integrity of student grades is not maintained, the quality of the institution concerned is questioned. The students' integrity-checking scheme developed in this study has been implemented into a software prototype. The software prototype compares students hash values before they are changed to the students grades after they have been changed by the administrator, lecturer or unknown parties. With the prototype of the software, it has been proven that if a change in the value of students is done illegally, the leader gets a pop-up notification. With this mechanism, the quality of institutions can be maintained, one of which is to improve student data security and prevent fraud from all parties.

REFERENCES

- [1] Jean-Philippe Aumasson (2018), *Serious Cryptography Practical Introduction to Modern Cryptography*, No Starch Press San Francisco
- [2] Ratna Anak P.R, Shaugi Ahmad, Purnamasari Prima D., Salman Muhammad, (2013), *Analysis and Comparison of MD5 and SHA-1 Algorithm Implementation in Simple-O Authentication based Security System*, IEEE.
- [3] Minematsu Kasuhiko (2018), *Breaking Message Integrity of an End-to-End Encryption Scheme of LINE*, University of HyogoKobeJapan
- [4] Liu Hongjun, Kadir Abdurahman, Liu Jian, (2018), *Keyed hash function using hyper chaotic system with*

time-varying parameters perturbation, IEEE Access open access journal.

[5] Piyush Gupta, Sandeep Kumar, (2014), "A Comparative Analysis of SHA and MD5 Algorithm", *International Journal of Computer Science and Information Technologies*, Vol. 5 (3).

[6] M. Sumagita and I. Riadi, "Analysis of Secure Hash Algorithm (SHA) 512 for Encryption Process on Web Based Application," vol. 7, no. September, pp. 373–381, 2018.

[7] S. Gueron, S. Johnson, and J. Walker, "Sha-512/256," *Proc. - 2011 8th Int. Conf. Inf. Technol. New Gener. ITNG 2011*, pp. 354–358, 2010.

[8] T. Grembowski *et al.*, "Comparative analysis of the hardware implementations of hash functions SHA-1 and SHA-512," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 2433, pp. 75–89, 2002.