Study and Comparative Analysis of Different Hash Algorithm

Kamlesh kumar Raghuvanshi
Purnima Khurana
Purnima Bindal

Abstract
In today’s world, security is the main issue during transmission or stored the data. Integrity is one of the main parameter to ensure the security over data. There are many hash algorithms proposed to ensure the integrity but almost all the algorithms have proven breakable or less secure. In this paper authors have review all such algorithms. Authors have implemented these algorithms and compare them on the basis of time, avalanche effect and space.

Keywords: Computer Security, SHA, Hash, Message Digest, Integrity

Introduction
A cryptographic hash function is used to ensure the integrity of the transmitted data or stored data. Sometimes it is also called digest of a message. Hash function generates a fixed size message digest of a given message; this message digest is treated as a signature of that message. Hash function can be defined mathematically as \( MD = H_F(M) \), where \( H_F \) is a hash function hold the following properties [9].

a. It is a one way function means it is easy to calculate MD from M but it is impossible to calculate M from MD.
b. It should be difficult to find to such messages M1 and M2 which generates same message digest i.e. \( H_F(M1) \neq H_F(M2) \).

There are many algorithms designed to implement the hash function. MD-2, MD-4, MD-5, SHA-0, SHA-1 and SHA-2 are the best known algorithms for message digest. After this many researchers have also proposed their own algorithms for the same such as SHA-192. In the next section authors have discussed all these algorithms in detail.

Study of Hash Algorithms
In this section authors have provide brief introduction about all the existing hash algorithms.

MD-2
It is a cryptographic hash algorithm which generates a message digest of 128 bits. It was published in August 1989. It takes 18 rounds of its compression function to generate a 128 bit digest. [9]

In 2004, MD2 was shown to be vulnerable to a preimage attack with time complexity of \( 2^{73} \) compression function evaluations.

In 2009, MD2 was shown to be vulnerable to a collision attack with time complexity of \( 2^{63.3} \) compression function evaluations.

MD-4
It is another cryptographic hash algorithm, generates a fixed 128 bits message digest. It takes 48 rounds of its compression function. It was published in 1990 [9].

A collision attack published in 2007 can find collisions for full MD4 in less than 2 hash operations.

MD-5
MD-5 generates a message digest of fixed 128 bits. It takes 64 rounds. It was published in 1992. [6]

A 2013 attack by Xie Tao, Fanbao Liu, and Dengguo Feng breaks MD5 collision resistance in \( 2^{18} \) times.

SHA-0
SHA-0 belongs from SHA family; it is another cryptographic hash algorithm generates a message digest of fixed 160 bits. It takes 80 rounds. It was published in 1993[7].

A 2004 attack by Bihamet Al breaks SHA-0 collision resistance at \( 2^{51} \).

SHA-1
SHA-1 generates a message digest of 160 bits. It takes 80 rounds and was published in 1995. It is the most widely used algorithm for integrity. Reason for its popularity among existing algorithms is its time efficiency and its robustness. [2]

Later on, a 2011 attack by Marc Stevens can produce hash collisions with a complexity of \( 2^{89} \) operations.
SHA-2
SHA-2 is a collection of different hash functions i.e. SHA-224, SHA-256, SHA-384 and SHA-512. None of them have proven completely breakable but still these algorithms are not preferred to ensure the integrity because they are not time efficient as SHA-1.[9]

It is found that, none of the hash algorithm is secure to ensure the integrity except SHA-2 but it is found that it is not time efficient. Many researchers have found these problems and proposed their own algorithms as a solution.

SHA-192[2]
SHA-192[2] is the hash algorithm proposed in 2009. The authors of SHA-192[2] have proposed its own compression function which is similar to SHA-1, the only difference in SHA-1 and SHA-192[2] is that SHA-192 uses 6 chaining variable of 32 bits in its compression function which generates 192 bits output. The compression function of SHA-192[2] is shown in Figure 1.

Here, A, B, C, D, E, F is the chaining variable. Each chaining variable holds 32 bits information. Initially all the chaining variable initialized with some value and during processing it changes its value and holds processing results and at last generates a result of 192 bits message digest.

SHA-192[1]
SHA 192[1] is another hash algorithm proposed in 2013. In this authors have proposed a new compression function to generate a message digest of 192 bits. Authors have combined the compression function of MD-5 and SHA-192 and take 64 rounds of compression function for each 512 bits message block. Compression function of SHA-192[1] is shown in Figure 2.

Comparative Analysis
In this section, authors have compared all the algorithms with each other. As discussed earlier, there are many algorithms that are used to ensure integrity over data but most of them are proven breakable. Comparison of all the algorithms discussed in Section II is shown in Table 1.

Table 1 Comparison between different hash algorithms on basis of general properties.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Output Size</th>
<th>Rounds</th>
<th>Collision Found</th>
</tr>
</thead>
<tbody>
<tr>
<td>MD-2</td>
<td>128</td>
<td>18</td>
<td>YES</td>
</tr>
<tr>
<td>MD-4</td>
<td>128</td>
<td>48</td>
<td>YES</td>
</tr>
<tr>
<td>MD-5</td>
<td>128</td>
<td>64</td>
<td>YES</td>
</tr>
<tr>
<td>SHA-0</td>
<td>160</td>
<td>80</td>
<td>YES</td>
</tr>
<tr>
<td>SHA-1</td>
<td>160</td>
<td>80</td>
<td>YES</td>
</tr>
<tr>
<td>SHA-2</td>
<td>224/256/38</td>
<td>64/80</td>
<td>Theoretical</td>
</tr>
<tr>
<td>SHA-192[2]</td>
<td>192</td>
<td>80</td>
<td>No</td>
</tr>
<tr>
<td>SHA-192[1]</td>
<td>192</td>
<td>64</td>
<td>No</td>
</tr>
</tbody>
</table>

No one has completely found collision on SHA-2 algorithms, but still SHA-2 algorithms are less preferable compare to SHA-1 because SHA-2 is not time efficient as SHA-1.
Also some authors have developed their own SHA algorithms by doing some modification on compression function named SHA-192.

To check the efficiency and strength of these algorithms, authors have developed these algorithms and compare it with SHA-1. Dot Net implementation has used to implement these algorithms. Intel Pentium Dual Core E2200 2.20 Ghz, 1 GB of RAM and Window-XP SP2, have used in which performance data is collected.

**Time Analysis**

An experimental result of time taken by each algorithm for generating message digest is shown in Table 2

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1KB</td>
<td>0.015</td>
<td>0.147</td>
<td>0.068</td>
</tr>
<tr>
<td>5KB</td>
<td>0.182</td>
<td>1.501</td>
<td>0.503</td>
</tr>
<tr>
<td>10KB</td>
<td>0.655</td>
<td>2.605</td>
<td>1.288</td>
</tr>
</tbody>
</table>

It is clearly seen that time taken by SHA-1 is very less compared to other algorithms hence it can be said that SHA [1] and SHA [2] are not time efficient algorithms. Figure 3 shows the graphical representation of Table 2.

![Figure 3 Timing comparison between SHA-1, SHA-192[2] and SHA-192[1]](image)

From Table 3, it is clearly seen that avalanche effect of SHA-1 and SHA-192[2] are closed to 50% while SHA-192[1] is far away from 50%. Hence It can be said that internal structure of SHA-1 and SHA-192[2] are more robust than SHA-192[1].

**Conclusion and Future Work**

This paper gives the over view about all the integrity algorithms. It is found that almost all the integrity algorithms have proven breakable except SHA-2 but it is not time efficient. Many researchers have proposed their own algorithms but none of them are time efficient as SHA-1 and also there are chances of improving the internal strength of these algorithms.

Future work can be done on this to reduce the time delay and also some work can be done to improve the internal strength of this algorithm.

**References**


