The Effect of Cryptographic Hash Functions on Speed and Uniqueness

Nitin Kanuri

Mills E. Godwin High School

Abstract

The role of hash functions is essential to the inner workings of modern cryptography and technology. With the emergence of newer and faster hashing functions, there must be a method to compare them and identify the most ideal hash. The purpose of this experiment is to test for the most supreme hash function based on performance of speed and uniqueness. The uniqueness is accurate reflection of the possibility of a collision. There are many reasons for the importance of the experiment, one being that for many companies, a breach in data protection could be unbearably costly. Modern costumers invest in companies that provide as much security as possible. This experiment was conducted in the Spring Tool Suite 4 (Eclipse) software and written in Java using a Bouncy House implementation to code and run the experiment. The experiment was run on a Linux operating system. The hash functions that were tested were MD5, SHA-256, Tiger, and Whirlpool. The research hypothesis was that SHA-256 would be the fastest and the most unique hash function. The results proved that the data was precise. When testing speed, it was shown that most of the data was not due to chance. However, for the majority of uniqueness aspect, the data was not significant. With this experimentation and future research, hash functions will grow so advanced that the probability of collisions and breaches will become close to impossible.

Introduction

Hash functions consist of a broad spectrum of functions. The easiest way to explain a hash function is by describing its actions. The cryptographic hash function converts a message/ input into a hash value of fixed-length depending on the individual function. What makes a hash function unique its inability to be decrypted and finding two inputs with the same hash value, which is known as a collision. Without using the proper hash function, data can be susceptible to: "information exposure, trojans, viruses, and network attacks" (Kehe 2015). This experiment specifically tests cryptographic hash functions. There are multiple properties that lay the boundaries for a function to be considered a cryptographic hash function: it is deterministic, should not have two different inputs should not yield the same digest (output), and that there should be an avalanche effect in where a small change to the input should completely change the output in a way that is has no relation to the original output.

Four hash functions, the four levels of independent variables , are being tested: MD5, ShA-256 (control), Whirlpool, and Tiger. The experiments tests which hash function is the fastest and is the most unique which are the two dependent variables. The most unique will be tested using a uniform distribution on how many "1" digits it contains. This digit was randomly selected using a random number generator. Although it the method of testing is through arbitrary means, a clear distribution will be shown (Molina 2005). The research hypothesis is that if the fastest and most unique hash function is tested for, then SHA-256 will be the fastest and most unique. The reason behind the research hypothesis is because the SHA family is the most commonly used and trusted family of hash functions.

Hash functions are computationally intensive and each chosen hash function (level of IV) is unique to its purpose. There is no method to amplify or improve the performance of a hash

function. SHA-256 was chosen as a hash function that is part of the SHA (Secure Hash Algorithm) Family. The SHA family is the norm of accepted standards in cryptography and published by the National Institute of Standards and Technology . SHA-256 is popularly used for mining for security and privacy. Another one of the hash functions being tested is Tiger. There are no known attacks on the Tiger Hash Function. This is important because that means this function is reliable and has an extremely low chance of being exploited. Unlike traditional hashes, Whirlpool has a Miyaguchi-Preneel scheme (one way) instead of a Davies-Meyer scheme. Whirlpool was created in 1996, therefore it has time to accumulate advancements (Mendel 2007). The main function of Whirlpool is to check integrity and generate digital signatures. Whirlpool has many similarities with the Advanced Encryption Structure. Therefore, the advantages also carry on: less memory during implementation, increased efficiency in performance, and hardware convenience (Kehe 2015). MD5 is the last level of IV being tested. The main purpose of MD5 is to authenticate digital signatures. The ability of each function is unique to itself and all the functions have major differences between each other. The levels were chosen based on the properties that varied the most. Hash functions cannot be amplified or improved to increase the performance or efficacy.

Procedure

A computer with a fairly fast processor was used in case the program crashes. The code was written in the Spring Tool Suite (Eclipse) for each hash function and was developed using a Bouncy Castle implementation. Twenty-five words used as the twenty-five trials were found using a random word generator. The same implementation was used for the all the hash functions tested. Four different cryptographic hash functions, the independent variables, were tested: MD5, SHA-256, Tiger, and Whirlpool. The code was written to find the digest and the time (milliseconds) taken to find it. The SHA-256 function is the control of the experiment as it is the most commonly used hash. The output of speed found in the console is shown in milliseconds, but it is recorded as the millisecond value divided by the characters of digest which varies for each trial. A random number generator of zero through nine was used and the number "1" was received. Using the output of the digest found in the console, the number of "1" in the digest is found for each trial. Using that data value received, the standardized value was calculated by using the self-derived formula $\frac{x}{chars\ of\ digest} * 48$. The characters of digest are based on the output of each hash function (32 for MD5, 64 for SHA-256, 48 for Tiger, and 128 for Whirlpool). The resultant that was multiplied by forty-eight because the value was found by averaging the characters of digest for all the indicated hash functions $\frac{32+64+48+128}{4} = 48$. The standardized value was recorded to find the uniqueness of the hash function, although the regular # of 1 value was not recorded as it has no relevance. The speed in milliseconds per byte and the uniqueness of each function are the two dependent variables that are tested for. Inferential statistics was done on the recorded data.

Results

The effect of hash function on speed and uniqueness was analyzed and results of the statistical analysis is shown in table 3 and 4 respectively. A research hypothesis was developed that when testing for the fastest and most unique hash functions, then SHA-256 will be the fastest and the most unique. The mean for each independent variable was calculated for speed and uniqueness. When analyzing the speed, the mean of MD5 was 6.43 ms/byte, the mean for SHA-256 was 9.26 ms/byte, the mean for Tiger was 8.33 ms/byte, and the mean for Whirlpool is 24.28 ms/byte. When analyzing the uniqueness, the mean of MD5 was 3.36, the mean of SHA-256 is 3.25, the mean of Tiger was 4.16, and the mean of Whirlpool is 3.15. MD5, with the lowest average of ms/bytes, had the fastest speed, and SHA-256, with the lowest average of one digits, ended up being the most unique. Due to these results, the research hypothesis was not supported by the data, but SHA-256, as stated, was the most unique. The SHA-256 hash function uses modern norms of cryptography and especially in encryption, its efficiency is remarkable (Zhu 9). The slow speed of Whirlpool is understandable. It is an especially computationally intensive function which has a low collision rate that might be due to the low speed (Zalewski 16). The standard deviations for the independent variables of speed and uniqueness were low, indicating the data was tight and precise. The normal distribution graph of the data, Figure 2, sums the results in a precise way and shows how often the "1" value is repeated.

A t-test was performed on the received data using a level of significant of 0.05 and degrees of freedom of 24. The null hypothesis was that there would be no statistically significant different in speed and uniqueness of hash functions. Many t-tests were calculated and when the t-value is lower than the critical table value, it meant that the data was not significant and most

likely due to chance. This occurred often when testing uniqueness as shown in table 4, but rarely when testing speed, as shown in table 3.

Conclusion

The purpose of this experiment was to determine the effect of hash function on speed and uniqueness. The hash functions were tested to four levels of the independent variable: MD5, SHA-256 (control), Tiger, and Whirlpool. Each hash function was run and timed in a software setting to measure the dependent variables: speed and uniqueness. A research hypothesis was formulated that if the hash functions are tested for their properties, then SHA-256 will be the fastest and the most unique. A t-test was performed on the data. When testing speed, it was found that the majority of the data was significant, in contrast, when testing uniqueness, it was found that the majority of the data was not significant. This implies that the results for speed was simply not due to chance, but when testing uniqueness, they most likely were due to chance.

There are many possible explanations for the results of this experiment. The SHA-256 was the control and is the what all the hashes are compared with. The reason SHA-256 is popularly used is because there has been no known attack on the SHA-2 family (Mouha 1). I believe that the data that was received for the speed was statistically significant because there is a comparative difference in speed of the four hash functions. The reason that the data for uniqueness was not statistically significant might have been due to extremely high probabilities of a collision. The uniqueness for all the hash functions were so high that the probability of a collision is either unusually rare or close to impossible. There are many sources of research that can be to compare the data of this experiment with. While publishing hash functions is common, hash functions are commonly compared for their properties. However, the process of finding bugs in cryptographic hash function implementations is equally important.

There were sources of error in this experiment and the experiment could have been improved in a multitude of ways. Although there was no human error involved, there could have been more trials and more bytes being tested to get the most accurate data. Possibilities of future research includes testing more hash functions at the same time and include a hash function from all the major hash function families to find the one hash to rule them all.

References

Peer-Reviewed

Kehe Wu, Yi Li, Long Chen, Zhuxiao Wang. (2015) Research of Integrity and Authentication in OPC UA Communication Using Whirlpool Hash Function. *Applied Sciences* 5:3, pages 446-458.

Mendel, Florian & Rijmen, Vincent. (2007). Cryptanalysis of the Tiger Hash Function. 536-550. 10.1007/978-3-540-76900-2_33.

Molina, Mayra & Niccolini, Saverio & Duffield, Nick & Dante, C. (2005). A comparative experimental study of hash functions applied to packet sampling.

Mouha, N., Raunak, M. S., Kuhn, D. R., & Kacker, R. (2018). Finding Bugs in Cryptographic Hash Function Implementations. IEEE Transactions on Reliability, 67(3), 870–884. doi: 10.1109/tr.2018.2847247

Zalewski, P. (2008). FPGA design and performance analysis of SHA-512, whirlpool and PHASH hashing functions.

Zhu, S., Zhu, C., & Wang, W. (2018). A New Image Encryption Algorithm Based on Chaos and Secure Hash SHA-256. Entropy, 20(9), 716. MDPI AG. Retrieved from http://dx.doi.org/10.3390/e20090716

Non-Peer Reviewed

Secure Hash Algorithms. (2019, November 5). Retrieved January 9, 2020, from https://en.wikipedia.org/wiki/Secure_Hash_Algorithms.

Appendix

Table 1. Raw Data Table for The Effect of Hash Function on Speed

Trial	Word/Bytes	Hash Functions			
		Speed of MD5	Speed of SHA-256	Speed of Tiger	Speed of Whirlpool
1	Amount/6	8.333333333	16.5	8.333333333	30.5
2	Carve/5	6	13.6	9.6	31.6
3	Aloof/5	6.4	8.8	5.8	38.5
4	Sling/5	8	9.8	11.4	28.6
5	Innate/6	5.833333333	10	5.5	22.66666667
6	Follow/6	4.5	7.333333333	11.83333333	19
7	Balloon/7	6	7	8.714285714	34.67
8	Distinct/8	3.25	6	5.375	15.875
9	Quickest/8	3.75	5.875	8.375	14.75
10	Quarrelsome/11	2.363636364	3.909090909	3.909090909	9.727272727
11	Signal/6	4.333333333	7	4.5	18.16666667
12	Permit/6	9.333333333	7.833333333	4.666666667	19.83333333
13	Bear/4	8	12	16.5	27.5
14	Seat/4	11.25	13	9	27.75
15	Slam/4	12.5	11	8.75	30.25
16	Prevent/7	3.714285714	6.142857143	3.857142857	24.669
17	Oatmeal/7	3.428571429	8.714285714	3.857142857	20

18	Chess/5	5	9.6	5.4	23.4
19	Twist/5	6	8.6	5.4	27.6
20	Spin/4	7	10.75	6.75	28.5
21	Bloody/6	8.166666667	11.83333333	5.666666667	20.16666667
22	Reply/5	7.2	8.6	5.4	23.4
23	Dull/4	6	10.5	6.75	28.5
24	Sheet/5	4.8	10	5.4	21.8
25	Dinner/6	9.666666667	7	4.5	19.66666667
Average		6.432926407	9.255649351	8.333333333	24.28365091

Table 2. Raw Data Table for The Effect of Hash Function on Uniqueness

Trial	Word/Bytes	Hash Functions			
		Standardized Value of "1's" for MD5	Standardized Value of "1's"for SHA-256	Standardized Value of "1's" for Tiger	Standardized Value of "1's" for Whirlpool
1	Amount/6	7.5	6	6	2.625
2	Carve/5	3	3	4	2.625
3	Aloof/5	6	3.75	1	4.125
4	Sling/5	1.5	3.75	3	4.125
5	Innate/6	1.5	3.75	3	3.75
6	Follow/6	1.5	3	6	4.125
7	Balloon/7	3	2.25	5	3
8	Distinct/8	0	3.75	3	1.125

9	Quickest/8	4.5	3	5	1.5
10	Quarrelsome/11	4.5	3	1	4.875
11	Signal/6	4.5	3.75	6	2.625
12	Permit/6	4.5	3	4	4.125
13	Bear/4	3	2.25	4	3.375
14	Seat/4	1.5	3.75	4	3
15	Slam/4	4.5	5.25	6	3
16	Prevent/7	1.5	3	5	2.25
17	Oatmeal/7	0	4.5	4	3.375
18	Chess/5	1.5	3.75	4	3
19	Twist/5	9	0.75	9	2.625
20	Spin/4	1.5	3.75	5	2.25
21	Bloody/6	4.5	4.5	5	2.25
22	Reply/5	3	2.25	2	3.375
23	Dull/4	3	2.25	3	3.75
24	Sheet/5	6	0.75	4	3.75
25	Dinner/6	3	2.25	2	4.125
Average		3.36	3.24	4.16	3.15

Table 3. Statistical Analysis for the Effect of Hash Function on Speed

Descriptive Info	Hash Function			
	MD5	SHA-256	Tiger	Whirlpool
Mean	6.43 ms/bytes	9.26 ms/byte	8.33 ms/byte	24.28 ms/byte
Range	10.14 ms/byte	12.59 ms/byte	12.64 ms/byte	28.78 ms/byte
Min	2.36 ms/byte	3.91 ms/ byte	3.86 ms/byte	9.72 ms/byte
Max	12.50 ms/byte	16.50 ms/byte	16.50 ms/byte	38.5 ms/byte
Variance	6.48	7.94	9.13	43.79
Standard Deviation	2.55	2.82	3.02	6.62
1 SD	3.88 – 8.98	6.44 – 12.08	5.31 – 11.35	17.66 – 30.9
2 SD	1.33 – 11.53	3.62 – 14.9	2.29 – 14.37	11.04 – 37.52
3 SD	0 – 14.08	0.8 – 17.72	0 – 17.39	4.42 – 44.14
Number	25	25	25	25

Results from t-test

MD5 vs SHA-256 t= 3.716 $p < 0.05$	SIGNIFICANT
MD5 vs Tiger t=0.730 $p > 0.05$	NOT SIGNIFICANT
MD5 vs Whirlpool t=12.588 $p < 0.05$	SIGNIFICANT
SHA-256 vs Tiger t=2.718 $p < 0.05$	SIGNIFICANT
SHA-256 vs Whirlpool t=10.447 $p < 0.05$	SIGNIFICANT
Tiger vs Whirlpool t=11.873 $p < 0.05$	SIGNIFICANT

$df = 24$, $\alpha = 0.05$, $t = 2.011$ for significance

Table 4. Statistical Analysis for the Effect of Hash Function on Uniqueness

Descriptive Info	Hash Function			
	MD5	SHA-256	Tiger	Whirlpool
Mean	3.36	3.25	4.16	3.15
Range	9	5.25	8	3.75
Min	0	0.75	1	1.125
Max	9	6	9	4.875
Variance	4.93	1.44	3.14	0.81
Standard Deviation	2.22	1.20	1.77	0.9
1 SD	1.14 – 5.58	2.05 – 4.45	2.39 – 5.82	2.25 – 4.05
2 SD	0 – 7.8	0.85 – 5.65	0.62 – 7.48	1.35 – 4.95
3 SD	0 – 10.02	0 – 6.85	0 – 9.14	0.45 – 5.85
Number	25	25	25	25

Results from t-test

MD5 vs SHA-256 t= 0.238 p > 0.05	NOT SIGNIFICANT
MD5 vs Tiger t= 1.408 p > 0.05	NOT SIGNIFICANT
MD5 vs Whirlpool t=0.438 p > 0.05	NOT SIGNIFICANT
SHA-256 vs Tiger t=2.149 p < 0.05	SIGNIFICANT
SHA-256 vs Whirlpool t=0.300 p > 0.05	NOT SIGNIFICANT
Tiger vs Whirlpool t=2.541 p < 0.05	SIGNIFICANT

df = 24, $\alpha = 0.05$, t = 2.011 for significance

Figure 1. The Correlation between Hash Function and Speed in Milliseconds per Byte

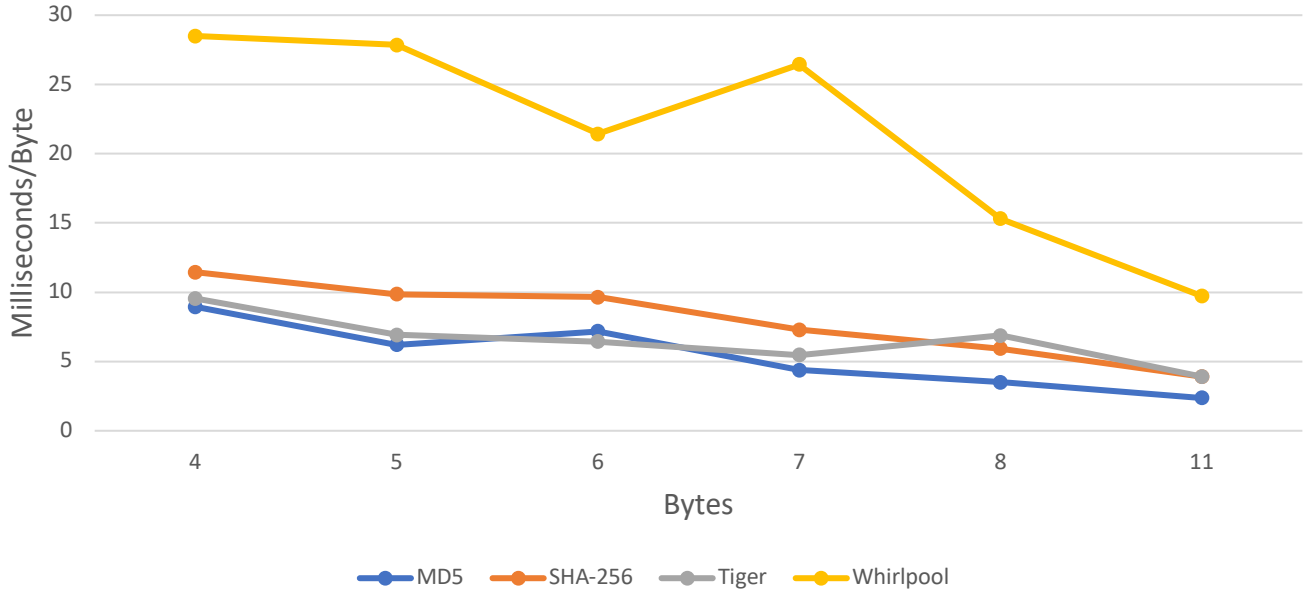
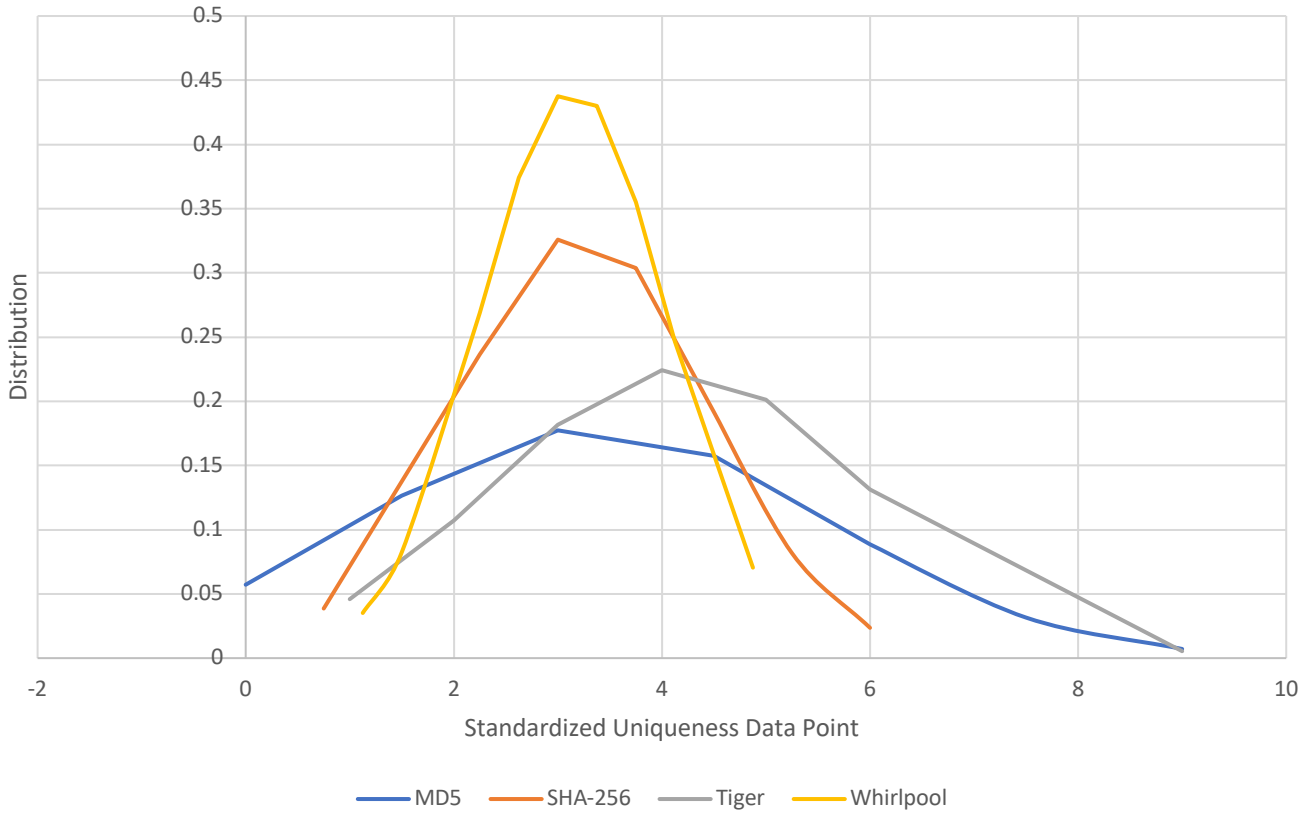


Figure 2. Normal Distribution of the Correlation between Hash Function and Uniqueness of Digest



EDD

Title: The Effect of Cryptographic Hash Function on Speed and Collisions

SHA-256	Tiger	Whirlpool	MD5
25 words	25 words	25 words	25 words

Constant: Same operating system, sample implementation, same input

DV: Speed measured in milliseconds per character. Uniqueness measured in standardized repetition of "1" values.