Measuring Hash Trustworthiness via Collision Utility Metrics: Logical Cryptanalysis of MD4

Alexander M. Scheel
Iowa State University
Ames, IA, USA
Email: alexander.m.scheel@gmail.com

Eric W. D. Rozier
Iowa State University
Ames, IA, USA
Email: erozier@iastate.edu

***Abstract*—The discovery of fast collision attacks in cryptographic hash functions has traditionally resulted in the immediate deprecation of that hash function. In this paper we propose five scalable and practical metrics for evaluating the utility of collision classes based on boolean constraints and show that the published attacks by X. Wang, Y. Sasaki, P. Kasselman, H. Dobbertin, and M. Schlaffer in MD4 have high utility. We expand on existing attacks by developing a series of techniques based on logical cryptanalysis to find over 35,000 collisions in MD4 based on existing collisions, through the novel definition of a collision neighborhood. We demonstrate new techniques for inductively building full collisions from reduced round variants of MD4. We propose these techniques as a mechanism for measuring hash trustworthiness and discuss potential applications to real-world systems.**

I. Introduction

Cryptographic hash functions form the core of many protocols. From file integrity checks to verify long term storage of data, to cache invalidation techniques, and use as a building block in network protocols such as Kerberos and TLS, this class of functions necessarily has some strong guarantees about the properties of these functions. There are three basic properties hash functions must have to be considered cryptographically secure:

- Preimage Resistance: It should be computationally hard to find the inverse of a hash function.
- Second Preimage Resistance: Given an input block, it should be computationally hard to find a second block which hashes to the same value as the first block.
- Collision Resistance: It should be computationally hard to find two blocks which hash to the same value.

Note that a second preimage is necessarily a collision and is also a stronger result.

From an attacker's perspective, finding a preimage or second preimage has been difficult. To the author's knowledge, the current bound for a preimage attack in MD5 is $2^{123.4}$ by Y. Sasaki and K. Aoki [1]. Further, while J. Kelsey and B. Schneier proposed a method for finding second preimages in less than 2^n, we note that this requires rather rather long messages [2]. However, collision attacks are well within reach for adversaries, and well past only theoretical attacks. The work of H. Dobbertin [3], X. Wang [4], M. Schlaffer [5] and others demonstrate the ease with which collisions can be found.

From a cryptographic perspective, the existence of a feasible collision attack breaks the requisite properties, thus deprecating the function's use. In some instances however, there continues to be widespread use of deprecated hash functions. One widespread example is the continued use by Git of SHA-1, despite M. Stevens et al. having published the first known full SHA-1 collision in February of 2017 [6]. However, this attack had a bound of $2^{63.1}$ – faster than the theoretical 2^{80} but still requiring significant investment in compute resources.

We define a series of criteria for analyzing the utility of a collision class. Collision classes of high utility are closer to a second preimage attack in the amount of flexibility they provide to an attacker, whereas classes of low utility may not affect all systems which use a particular hash function. Furthermore, we demonstrate new techniques for working with collisions under a logical cryptanalysis framework. Throughout this paper, we make use of the MD4 hash function due to the ease of generating new classes of collisions.

II. Terminology & Notation

TODO.

$\mathcal{C}$ for collision class. Ordered tuple, indexable. $\mathcal{F}$ for family of collision classes. Ordered set, indexable. $[i]$ is an indexing function. R is number of rounds of F or C.

III. Measuring the Utility of a Collision

We propose the following metrics for evaluating the utility of a collision class:

1) The number of unique differentials a collision class has.
2) The number of unit-step neighbors a collision class has.
3) The maximum count of zeros in a the binary representation of a colliding block (and likewise with ones).
4) Whether there exists a block which collides under multiple initial values.
5) Whether or not zero, one, or both of the blocks in a collision may be of ASCII values under any input block difference.

Note that the first three are quantitative measures providing some measure of flexibility of a collision class, whereas the latter two are merely looking for a single witness for having the property. Depending on the scenario, specific properties of a collision may be of more interest than others.

The first metric evaluates the flexibility of the differential path. A differential path with more flexibility will have more differentials which produce blocks with the given differential path. More differentials implies a greater flexibility in choice of colliding block, and possibly allowing for multiple collisions for a given colliding block. Furthermore, a collision with more differentials is more likely to satisfy the last metric, having a pair of blocks–both ASCII–which produce a collision.

The second metric evaluates the density of the neighborhood of a collision class. If a collision resides in a dense neighborhood, it provides more possible collision classes to search for a second preimage, chosen prefix, or other structure desired in a collision. If however, a collision class has no neighbors, then it cannot be used to find other possible classes for other input blocks.

The maximum quantity of zeros (or ones) in the binary representation of a block serve as a measure of the extremes to which a collision can be pushed. This can additionally be extended to any suitable bit pattern in any base to provide a more relevant metric as desired by the system under study.

The fourth metric evaluates the utility of a collision when the internal state of the hash function is unknown. If a collision occurs under multiple initial values, this could be used to attack some systems where user provided input is appended to unknown data and then hashed. If a block collides under a suitably large number of initial values, the attack becomes highly likely to occur successfully. However, measuring the exact number of initial values a block collides under is beyond the scope of this work.

The last metric is similar to the third in that it looks for specific bit patterns in a collision. ASCII is one example of a widely used constraint system. Further examples, such as JSON, XML, etc., may likewise be supplemented based on the specifics of the system.

If a collision class satisfies many of these properties, then it is more flexible and thus more likely to be used to target deployed systems. If, however, a collision class does not satisfy these properties, it impact is likely severely limited in scope, and may not provide useful information to find other collision classes which have higher utility.

IV. Techniques for Logical Cryptanalysis

The following techniques have been extensively tested on MD4 and partially tested on MD5 and believe to apply fully to MD5. They may or may not apply to any later hash function, such as SHA-1, SHA-2, or SHA-3, and have not been tested yet.

In the following sections, we use the collisions of X. Wang [7], Y. Sasaki [8], M. Schlaffer [5], H. Dobbertin [3], and P. Kasselman [9] for examples.

A. Distance Metrics

We introduce a distance function, δ between collision classes by the number of differences in intermediate rounds deltas. That is, given two collision classes, $C_1, C_2 \in \mathcal{C}$:

$$\delta(C_1, C_2) = \left|\{i : C_1[i] \neq C_2[i]\}\right| \tag{1}$$

TABLE I
Distance between Existing Collision Classes

	X. W.	Y. S.	P. K.	H. D.	M. S.
X. Wang's	0	21	26	27	12
Y. Sasaki's		0	25	26	2
P. Kasselman's			0	1	25
H. Dobbertin's				0	26
M. Schlaffer's					0

We find justification for this metric in the existing literature on MD4: H. Dobbertin's [3] and P. Kasselman's [9] collision classes have distance 1 under this metric.

Refer to Table I for the distances between existing collisions in MD4.

We can define a similar distance function, Δ, between families of collision classes by cardinality of the symmetric difference in the two collision families. That is, given two families of collision classes, $F_1, F_2 \in \mathcal{F}$:

$$\Delta(F_1, F_2) = \left|(F_1 \cup F_2) \setminus (F_1 \cap F_2)\right| \tag{2}$$

This is convenient for when the specifics of the differential path do not matter, merely that there exist at least one collision class of the specified form.

B. Neighborhoods

We define the neighborhood of a collision class, $C \in \mathcal{C}$, to be the set of all other collisions at a fixed distance, $d \in \mathbb{N}$, from C. That is:

$$N(C, d) = \{C_j \in \mathcal{C} : \delta(C, C_j) = d\} \tag{3}$$
$$N(C) = N(C, 1) \tag{4}$$

For instance, $C_{Wang} \in N(C_{Sasaki}, 21)$. The distance parameter, d, may optionally be omitted, in which case the unit distance is implied. Thus, $C_{Kasselman} \in N(C_{Dobbertin})$.

The implications of collision class neighborhoods are discussed further in section V-B.

We can similarly define the neighborhood of a family of collision classes, $F \in \mathcal{F}$, to be the set of all other families of collision classes at a fixed distance, $d \in \mathbb{N}$, from F. That is:

$$\mathcal{N}(F, d) = \{F_j \in \mathcal{F} : \delta(F, F_j) = d\} \tag{5}$$
$$\mathcal{N}(F) = N(F, 1) \tag{6}$$

The distance parameter, d, may optionally be omitted, in which case the unit distance is implied.

Neighborhoods can be classified into three types: *expansion*, *internal*, and *mixed*. Let d be fixed. An *expansion* neighborhood of a collision class, $C \in \mathcal{C}$, is the neighborhood restricted only to those collision classes which only differ in rounds external to the collision family of C. An *internal* neighborhood of C is the neighborhood restricted only to those collision classes which differ in rounds internal to the collision family of C. An *mixed* neighborhood of C is the neighborhood restricted

TABLE II
NUMBER OF DIFFERENTIALS FOR EXISTING COLLISION CLASSES

Attack	**Size**	**Attack**	**Size**
X. Wang's	64	Y. Sasaki's	4
H. Dobbertin's	32	P. Kasselman's	32
M. Schlaffer's	64		

only to collisions which differ in a round internal and a round external to the collision family of C. That is:

$$N_{exp}(C,d) = \{C_j \in N(C,d) : \forall i \in F(C), C_j[i] = C[i]\}$$
$$N_{int}(C,d) = \{C_j \in N(C,d) : \forall i \notin F(C), C_j[i] = C[i]\}$$
$$N_{mix}(C,d) = \{C_j \in N(C,d) : \exists i \in F(C), C_j[i] \neq C[i] \text{ and } \exists k \notin F(C), C_j[k] \neq C[k]\}$$

C. Family Similarity

We define a relation among families of collisions across rounds. Let F_1 and F_2 be two different families of collisions. Then we say that F_1 and F_2 are *similar*, and notate it $F_1 \lesssim F_2$, if $R(F_1) \leq R(F_2)$ and $F_1 \subseteq F_2$. Note that, when $R(F_1) = R(F_2)$, this is equivalent to saying that F_2 is in some expansion neighborhood of F_1. Further, when $R(F_1) < R(F_2)$ and $F_1 = F_2$, then we say that F_2 is the *trivial extension* of F_1.

We claim that the following statements are true:

1) For every $F \in \mathcal{F}$, there exists F' such that $F' \lesssim F$ and $R(F') + 4 = R(F)$.
2) For every $F \in \mathcal{F}$, there exists F' such that $F \lesssim F'$.

TODO - Describe justification? Later sections?

D. Class Similarity

We claim that the aforementioned statements hold when considering individual collision classes instead of families of collision classes, but with less likelihood.

TODO - Describe technique applied neighborhood extensions

Give some reasoning for why it works, example graphs.

E. Miscellaneous

TODO - Describe chosen prefix, second preimage, ASCII, and other tips and tricks for faster model runs.

V. EMPIRICAL RESULTS

A. Unique Differentials

TODO Refer to Table II for the number of differential paths...

B. Unit-Step Neighborhood

TODO

Refer to Table III for the sizes of neighborhoods of existing collisions. In particular, note that Y. Sasaki's attack–which claimed to improve upon X. Wang's attack–has a larger neighborhood size, and likewise with P. Kasselman's attack which improved upon H. Dobbertin's attack.

TABLE III
NEIGHBORHOOD SIZES FOR EXISTING COLLISION CLASSES

Attack	**Size**	**Attack**	**Size**
X. Wang's	54	Y. Sasaki's	157
H. Dobbertin's	55	P. Kasselman's	60
M. Schlaffer's	100		

TABLE IV
ASCII BLOCKS IN EXISTING COLLISION CLASSES

Attack	**Single Block**	**Both Blocks**
X. Wang's	true	false
Y. Sasaki's	true	false
P. Kasselman's	true	**true**
H. Dobbertin's	true	**true**
M. Schlaffer's	true	false

C. Zeroes & Ones

TODO

Include graphic of count of max number of zeros/ones for all 35k collision classes.

Separate table for known attacks as data points. Examples as appendix.

D. Multicollisions

TODO

Include table detailing how many have multicollisions and how many don't. Hypothesis: all do since all existing data-points do.

Separate table for known attacks as data points. Examples as appendix.

E. ASCII Blocks

TODO

Include table detailing how many have ASCII blocks and how many don't.

See Table IV for results. Note that the validation of P. Kasselman's and H. Dobbertin's results do not hold for the latter attacks by X. Wang, Y. Sasaki, or M. Schlaffer. Thus, while the latter attacks have been viewed as being of better quality, under this particular metric, P. Kasselman's and H. Dobbertin's collision classes are better.

VI. NEW COLLISION

VII. BROADER IMPACTS

TODO - Discuss impacts on real-world systems like MD5, SHA-1, etc.

VIII. OPEN RESEARCH

The framework can be seen here **TODO**. Complete data set available upon request.

IX. CONCLUSION

The conclusion goes here.

ACKNOWLEDGMENT

The authors would like to thank the Department of Electrical and Computer Engineering and Nathan Mulbrook for providing hardware.

REFERENCES

[1] Y. Sasaki and K. Aoki, *Finding Preimages in Full MD5 Faster Than Exhaustive Search*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 134–152, https://www.iacr.org/archive/eurocrypt2009/54790136/54790136.pdf.

[2] J. Kelsey and B. Schneier, *Second Preimages on n-Bit Hash Functions for Much Less than* 2^n *Work*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, pp. 474–490, https://www.schneier.com/academic/paperfiles/paper-preimages.pdf. [Online]. Available: https://doi.org/10.1007/11426639_28

[3] H. Dobbertin, "Cryptanalysis of md4," *Journal of Cryptology*, vol. 11, no. 4, pp. 253–271, Sep 1998, https://link.springer.com/content/pdf/10.1007/s001459900047.pdf. [Online]. Available: https://doi.org/10.1007/s001459900047

[4] X. Wang, D. Feng, X. Lai, and H. Yu, "Collisions for hash functions md4, md5, haval-128 and ripemd," Cryptology ePrint Archive, Report 2004/199, 2004, http://eprint.iacr.org/2004/199.

[5] M. Schläffer and E. Oswald, *Searching for Differential Paths in MD4*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 242–261, https://link.springer.com/content/pdf/10.1007%2F11799313_16.pdf. [Online]. Available: https://doi.org/10.1007/11799313_16

[6] M. Stevens, E. Bursztein, P. Karpman, A. Albertini, and Y. Markov, "The first collision for full sha-1," Cryptology ePrint Archive, Report 2017/190, 2017, http://eprint.iacr.org/2017/190.

[7] X. Wang, X. Lai, D. Feng, H. Chen, and X. Yu, *Cryptanalysis of the Hash Functions MD4 and RIPEMD*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, pp. 1–18, http://www.infosec.sdu.edu.cn/uploadfile/papers/Cryptanalysis%20of%20the%20Hash%20Functions%20MD4%20and%20RIPEMD.pdf. [Online]. Available: https://doi.org/10.1007/11426639_1

[8] Y. Sasaki, L. Wang, K. Ohta, and N. Kunihiro, *New Message Difference for MD4*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007, pp. 329–348, https://link.springer.com/content/pdf/10.1007/978-3-540-74619-5_21.pdf. [Online]. Available: https://doi.org/10.1007/978-3-540-74619-5_21

[9] P. R. Kasselman, "A fast attack on the md4 hash function," in *Communications and Signal Processing, 1997. COMSIG '97., Proceedings of the 1997 South African Symposium on*, Sep 1997, pp. 147–150, http://www.mathcs.emory.edu/~whalen/Hash/Hash_Articles/IEEE/A%20fast%20attack%20on%20the%20MD4%20hash%20function.pdf.