

Forum: [Hack Forums](#)

Thread: [\[Tutorial\]Understanding Hash Algorithms](#)

Captain Crunch - July 24, 2011, 11:18 pm

I can't go over anything in a tutorial so I suggest you further your knowledge by gaining access to a book about cryptography.

Hash Functions:

Hash value is generated by a function H of the form:

$$h=H(M)$$

M is the variable length message and $H(M)$ the fixed length hash value. The hash function itself is not secret so means are taken to protect the hash value. An avalanche effect will occur. Inputs such as "hi" and "hi." will produce extremely different outputs.

According to Stallings the requirements for a hash function are:

1. H can be applied to a block of data of any size
2. H produces a fixed-length output.
3. $H(x)$ is relatively easy to compute for any given x , making both hardware and software implementations practical
4. For any given code h , it is computationally infeasible to find x such that $H(x)=h$. This is sometimes referred to as the one-way property.
5. For any given block x , it is computationally infeasible to find $y \neq x$ with $H(y)=H(x)$. This is sometimes referred to as weak collision resistance.
6. It is computationally infeasible to find any pair (x,y) such that $H(x)=H(y)$. This is sometimes referred to as strong collision resistance.

The first three being requirements for usable application of a hash to message authentication. The fourth being the one way property. The fifth assuring that a different message hashing to the same value as a given message cannot be found. The sixth referring to how resistant the hash function is to the birthday attack.

Operation of Simple Functions:

Input is viewed as a sequence of n -bit blocks. Input is processed one block at a time in a repetitive mode to produce an n -bit hash function.

Example of a simple hash function (I used the $_$ to indicate subscript)

$$C_i = b_{i1} \oplus b_{i2} \oplus \dots \oplus b_{im}$$

This is a very basic example. I won't be providing anything more complex, if you want to see one do some research.

Security of Hash:

Attacks on hash functions are grouped into brute force attacks and cryptanalysis. I personally like to extend the attacks to a third group: dictionary attacks. Please remember that this is not an official group so if you speak to a professor or anyone and say this they may correct you.

Brute Force Attacks:

You can find many definitions but I found the rsa's to be the most accurate.

"trying all (or a large fraction of all) possible values till the right value is found"

The strength of a hash function against this type of attack depends on the length of the hash code produced. For a code of n length the effort would be (I used $^$ to indicate superscript)

One way: 2^n

Weak collision resistance: 2^n

Strong collision resistance: $2^{(n/2)}$

As you can see brute forcing a hash code is extremely long!

Cryptanalysis:

Cryptanalysis attacks seek to exploit some property to perform an attack that isn't exhaustive.

Personally I find cryptanalysis on Hash functions to be extremely complex and difficult I myself don't understand it well enough to explain. Only if you are truly great at math and understand the structure of the function in detail do you stand a chance with this attack. I'll therefore just make a short summary of what the structure of a iterated hash function includes (This is the structure of popular hash's such as MD5):

- *Initial value

- *Chaining Variable

- *ith input block

- *compression algorithm

- *number of input blocks

- *length of hash code

- *length of input block

Dictionary Attack:

Once again remember that this isn't an official term nor is it the best suited term to use. I decided to include it since it is the most common way unprofessional crackers attempt to decrypt. Websites such as (<http://md5online.net/>) allow anyone with the knowledge of the type of hash to simply search a database that may have their hash stored. Of course this attack is useless with uncommon strings or long words. But with weak IV it is common to see the decryption.

"49f68a5c8493ec2c0bf489821c21fc3b" could simply be searched and the output of hi is returned to you!

Hash Algorithms:

I have decided to talk a bit about MD5 and SHA-1 because of their popularity and how often I see users wanting those type of hash's decrypted. I won't be talking about structure so much as that is rather long and complex. Instead I'll keep it short and sweet and talk about security.

MD5 Message Digest Algorithm:

Brief background: Developed by Ron Rivest at MIT. Formerly the most used secure hash algorithm. Currently it is broken and unsuitable for secure use.

MD5 takes in a random message of any length and puts out a 32 bit hexadecimal correspondent.

The strengths include:

- *every bit of the hash code is a function of every bit of input

- *results are well mixed

- *Brute force attacks can take an extremely long time

Weaknesses include:

- *It has been broken to say
- *cryptanalysis attacks are very effective
- *pseudocollision
- *Dobbertin's attack enabling a collision for the compression function

SHA-1:

Brief Background: Designed by the NSA. Most used SHA.

Takes in an input message less than a length of 2^{64} bits and produces a 40 bit hexadecimal output.

Strengths and Weaknesses:

- *Should be similar to MD5 since they are both derived from MD4

Should I use MD5 or SHA-1?

I would advise against using either but if we were to choose one or the other I would go with SHA-1.

- *Speed: MD5 wins this round.

- *Simplicity: Even

- *Brute force attacks: SHA-1 wins this round.

- *Cryptanalysis: SHA-1 wins this round.

- *Dictionary Attack: SHA-1 wins this round. This has nothing to do with its design but simply because there are many more resources for cracking MD5 with a database.

Thank you if you read this and please leave back some comment. If you found this helpful please tell me what I should write about next time. Thanks! Sorry for any errors, I'm finishing this up around 2 in the morning and really don't have the energy to check my grammar.

Credit Due where owned

NA. "MD5." Wikipedia. N.p., n.d. Web. 25 July 2011. <<http://en.wikipedia.org/wiki/MD5>>.

- - -. "SHA-1." Wikipedia. N.p., n.d. Web. 25 July 2011. <<http://en.wikipedia.org/wiki/SHA-1>>.

Smart, Nigel. Cryptography: An introduction. N.p.: n.p., n.d. Print.

Stallings, William. Cryptography and network security Principles and practice. Upper Saddle: n.p., n.d. Print.

- - -. Network Security Essentials. N.p.: n.p., n.d. Print.

Rdogg112 - July 25, 2011, 12:26 am

Nice tutorial bro. Very useful

Captain Crunch - July 25, 2011, 4:23 am

Thank you Rdogg112. I appreciate your feedback.

Basics. - July 25, 2011, 7:04 am

Nice tutorial on MD5. This tutorial will go far.

ZeKiiTo - July 25, 2011, 7:09 am

The best TUTO for the "Understanding of the the HASH" Thanks bro !! The best TUTO for the "Understanding of the the HASH" Thanks bro !!

Captain Crunch - July 25, 2011, 9:59 am

Thank you both that means a lot to me. Is there anything specific you'd like to see in a next thread?

Duncskunk - *July 25, 2011, 11:15 am*

good tut, it helped alot, thanx ;)

Captain Crunch - *July 25, 2011, 1:55 pm*

Thank you for the comment Duncskunk.