How can I reverse engineer a hash code?

I am building an application in C# that works with a Progress database. The passwords that are stored in this database are stored using a hash algorithm that Progress has not made public. However, I would like to authenticate using these hashes. Is it feasible to reverse engineer such a hash algorithm and how would I go about doing this?

To be clear: I am not looking for a way to get the unhashed passwords from the hashes. I'm looking for the algorithm to get a hash from a password.

hashing | reverse-engineering

asked Jul 31 '13 at 11:17

Pieter van Ginkel
222 2 10

First try to find a third party implementation of the hash. Perhaps somebody already did the work for you. If not, download a debugger/disassembler and get cracking. Good luck. – CodesInChaos Aug 1 '13 at 11:29

I also consider using non standard, closed source crypto without spec extremely negligent. What if they use a cheap unsalted hash? Many people think SHA-2(password) is a good choice. Consider using a standard hash in the application, instead of relying on the database for this. – CodesInChaos Aug 1 '13 at 11:31

Standard crypto answer: If it's undocumented, it's unsuited. Pretty similar to the medical world, actually. "No, we don't know what's in this syringe, but we're going to inject you with it. The manufacturer swears it will work". Run, quickly. – MSalters Aug 1 '13 at 13:20

@CodesInChaos: You are absolutely right that this is quite a horrible situation, but the whole point of the question is that I need this to be compatible. I'll probably migrate to SHA1, but for now I'm looking for compatibility. – Pieter van Ginkel Aug 1 '13 at 14:53

@Pieter Don't migrate to SHA-1 or even SHA-2. If you migrate, use a specialized password hash, like PBKDF2, bcrypt or scrypt. See How to securely hash passwords? on security.SE for details. – CodesInChaos Aug 1 '13 at 15:07

3 Answers

According to this topic you are not the first one who is looking for that algorithm. As already mentioned Progress is not willing to share much about this algorithm.

While it's most probably possible to do crack the algorithm, it's generally easier to use Progress itself to calculate hashes. The other option is to switch from using that Progress funtion to a more standard variants (like SHA-1).

Now, more clarification of what "variation" could possibly mean. Suppose something like:

```
string Encode(string pass) {
    string salt = "123456";
    return CRC16(pass + salt); // + for catenation
}
```

while it's still same CRC16 it will produce totally different results (comparing to `CRC16(pass)`). So the only way to get real algorithm is to use disassembler to see the actual code, since it's probably impossible to do anything observing only input and output.

There is no "structural approach" to guess algorithm from input and result unless you have a pattern, and that's *exactly* that ideal hash need not to have.

edited Aug 1 '13 at 4:36

answered Jul 31 '13 at 14:56

Petr Abdulin
477 3 11

Following links, this says it's a CRC-16. – AakashM Jul 31 '13 at 15:26

Programmers Stack Exchange works best with JavaScript enabled

Petr Abdulin Jul 31 '13 at 16:08

That's kind of my question: how does one go about figuring out what algorithm they are using. – Pieter van Ginkel Jul 31 '13 at 20:09

@Pieter I've updated my answer, hope that is what you are asking about. – Petr Abdulin Aug 1 '13 at 3:56

@PetrAbdulin Thank you for your feedback. I was able to extract the algorithm and it's now available at github.com/pvginkel/ProgressEncode. – Pieter van Ginkel Aug 3 '13 at 5:52

You could use progress to insert a hashed password into a temp var and compare that to the stored hash.

Using progress to make that hash.

answered Jul 31 '13 at 11:39

Pieter B
4,569 9 39

If nothing is known about how the hash algorithm works then this is probably the most straight forward method. That is unless you get lucky and find they are using unsalted MD5 hashes. In which case you probably don't want to use the algorithm anyway. – Tombatron Jul 31 '13 at 12:44

This does sound like a start of a process that I'm looking for. I'm just wondering where to go from there. Can you give some suggestions on a structered approach on actually working out the algorithm? – Pieter van Ginkel Jul 31 '13 at 20:08

I'm not really used to progress syntax but something like "SELECT ENCODE('TESTTHISPASSWORD') AS hashtocheck, passwordhash FROM usertable WHERE userid = 12345" and comparre both results. Check the links in the petr abdulin answer for more info. – Pieter B Aug 1 '13 at 8:17

Well, you could take a list of all hashing algorithms you can think of, run them against a list of the most popular passwords and check whether any of the results match any of your hashes. Of course this will only work if the site did not enforce a strong password policy and the hashing algorithm isn't exotic (or parameterized).

answered Jul 31 '13 at 11:39

Michael Borgwardt
27.8k 6 60 111

See the comment from @Petr. This probably won't work because even Progress themselves stated that it's a non standard hash function. – Pieter van Ginkel Jul 31 '13 at 20:06
