datenschutzbeauftragter-info.de

Hashwerte und Hashfunktionen einfach erklärt

Agnieszka Czernik

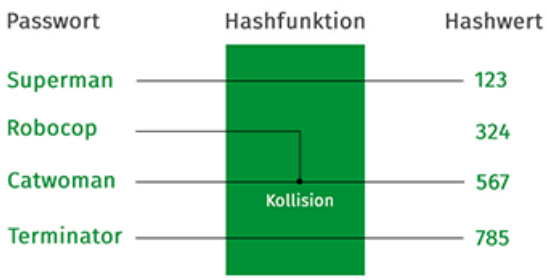
8-10 Minuten

Fachbeitrag

In der IT-Sicherheit ist oft die Rede von Hashwerten und Hashfunktionen, von MD5 oder SHA. Diese Begriffe werden auch in Zusammenhang mit Verschlüsselung erwähnt oder auch miteinander verwechselt. In diesem Artikel wird anhand eines Beispiels erklärt, was Hashwerte und Hashfunktionen sind und welche Anwendungsfelder diese haben.

Hashfunktion

Um die Erklärung einfacher zu machen, fangen wir mit einem Bild an:



Auf der linken Seite sehen wir 4 Passwörter von beispielsweise 4

Mitarbeitern eines Unternehmens. Die Hashfunktion (=Algorithmus) wandelt nun diese Passwörter in eine Zeichenfolge (dem Hashwert) mit einer festen Länge (hier 3 Zeichen) um. Für das Passwort „Superman“ bekommt man den Hashwert 123, dem Passwort „Robocop“ wird der Hashwert 567 zugeordnet, genauso wie dem Passwort „Catwoman“ und „Terminator“ bekommt 785.

Hashfunktionen reduzieren zunächst nur Zeichen beliebiger Länge (unterschiedliche Passwörter) auf Zeichen fester Länge (hier immer 3 Zeichen), sie werden also in eine kleine, kompakte Form gebracht.

Hashwert

Der Hashwert stellt das Ergebnis dar, welcher mittels einer Hashfunktion berechnet wurde. Man definiert eine feste Länge, wie lang ein Hashwert immer sein darf. Oft wird der Hashwert als eine hexadezimale Zeichenkette codiert d.h. der Hashwert besteht aus einer Zahlen und Buchstaben-Kombination zwischen 0 und 9 sowie A bis F (als Ersatz für die Zahlen 10 bis 15). Ein Hashcode aus 10 hexadezimalen Zeichen könnte so aussehen: „3d180ab86e“.

Eigenschaften

Eine Hashfunktion sollte die folgenden Eigenschaften haben:

- **Einwegfunktion**:
  Aus dem Hashwert darf nicht der originale Inhalt erzeugt werden können. In unserem Beispiel darf es nicht möglich sein, aus dem Hashwert „123“ den Ursprungstext „Supermann“ zu erzeugen.
- **Kollisionssicherheit:**
  Den unterschiedlichen Texten darf nicht derselbe Hashwert zugeordnet sein. Ist diese Voraussetzung erfüllt, so spricht man auch von kryptografischen Hashfunktionen. In unserem Beispiel liegt eine Kollision vor, da die Passwörter „Robocop“ und „Catwoman“ denselben Hashwert haben. Damit ist die Hashfunktion im Bild nicht kollisionssicher und es handelt sich nicht um eine kryptografische Hashfunktion.
- **Schnelligkeit:**
  Das Verfahren zu Berechnung des Hashwertes muss schnell sein.

Anwendungsfelder

Hashfunktionen reduzieren eine große Datenmenge auf eine kleinere Zeichenfolge. Darüber hinaus können Hashfunktionen als Integritätsschutz dienen, indem ein elektronischer „Fingerabdruck“ berechnet wird (hier spielt die erwähnte Kollisionsresistenz eine wichtige Rolle). Im Einzelnen seien beispielhaft die folgenden Anwendungsfelder erwähnt:

Prüfsummen

Lädt man sich ein Programm aus dem Internet runter, möchte man sicher gehen, dass es sich um das Original und kein manipuliertes Programm handelt, welches Schadsoftware enthalten könnte bzw. während des Download manipuliert wurde. Auch hier können Hashwerte hilfreich sein. Man berechnet einmalig den Hashwert des Original-Programmes (manche Hersteller stellen diesen zur Verfügung) und hat somit einen Fingerabdruck des Programms. Ist das Programm heruntergeladen, kann man erneut einen Hashwert mit derselben Hashfunktion berechnen und nun beide Hashwerte miteinander vergleichen. Abweichungen deuten darauf hin, dass es sich um eine manipulierte Version handelt. Hashfunktionen ermöglichen damit das Aufdecken möglicher Manipulationen.

Digitale Signatur

Bei der digitalen Signatur (das Pendant zur handschriftlichen Unterschrift) werden Hashfunktionen dazu verwendet, um „Fingerabdrücke“ von Nachrichten zu berechnen. Der Fingerabdruck wird zusammen mit der Nachricht an den Empfänger als Beweis der Integrität gesendet.

Die digitale Signierung funktioniert wie folgt:

1. Der Absender berechnet einen Hashwert aus seiner Nachricht.
2. Diesen Hashwert verschlüsselt er mit seinem privaten Schlüssel (=digitale Signatur) und übermittelt die Nachricht zusammen mit dem verschlüsselten Hashwert an den Empfänger.
3. Der Empfänger erstellt auf seiner Seite ebenfalls einen Hashwert der erhaltenen Nachricht (mithilfe derselben Hashfunktion).
4. Zudem entschlüsselt er den erhaltenen Hashwert mit dem öffentlichen

Schlüssel und vergleicht beide Werte miteinander.

Stimmen beide Werte überein, so kann der Empfänger davon ausgehen, dass die Nachricht bei der Übertragung nicht verändert wurde.

Wichtig: Es ist zu beachten, dass die digitale Signatur nicht die Nachricht verschlüsselt, sondern die Integrität der Nachricht verifizieren soll. Ist die Nachricht vertraulich, sollte diese zusätzlich verschlüsselt werden.

Speichern von Passwörtern

Passwörter sollen verschlüsselt gespeichert werden, damit bei einem Angriff diese nicht im Klartext gelesen werden können. Aber wie können die Passwörter genutzt werden, wenn diese „verschlüsselt“ gespeichert sind? Die Passwörter werden mittels einer Hashfunktion in Hashwerte umgewandelt – nur diese Hashwerte werden gespeichert (nicht die Passwörter selbst). Gibt ein Nutzer bei einer Authentisierung sein Passwort ein, wird bei der Eingabe wieder mit derselben Hashfunktion der Hashwert berechnet und mit dem gespeicherten Hashwert verglichen. Bei Übereinstimmung gilt der Benutzer als authentifiziert.

Stand der Technik

Die verwendeten Hashfunktionen sollten auf den neusten Stand der Technik sein. Welche Algorithmen bereits „geknackt“ wurden, findet man z.B. in dieser Liste.

Immer wieder hört man, dass Passwörter geknackt wurden. Betroffen waren z.B. Ashley Madison, LinkedIn und Dropbox. Aber wie können Passwörter geknackt werden, wenn man wegen der Einweg-Eigenschaften der Hashfunktionen nicht auf den ursprünglichen Text zurückschließen kann?

Zunächst muss man wissen, dass fast alle Algorithmen „offen“ liegen, diese also auch von Angreifern genutzt werden können. Das hat zur Folge, dass der Hashwertvon einem Passwort immer gleich ist egal ob es die Plattform oder der Angreifer berechnet.

> **Passwort:** „Superman“
> =
> **MD5-Hash:** 527d60cd4715db174ad56cda34ab2dce

Ein Angreifer kann sich also eine Liste mit typischen unsichere Passwörter erstellen und durch den Hashgenerator jagen. Wenn er nun die Datenbank mit den Hashwerten der Plattform stiehlt, kann er die Hashwerte mit seiner Liste vergleichen. Findet er in der geklauten Liste den Hashwert 527d60cd4715db174ad56cda34ab2dce, so weiß er, dass dieser Hashwert dem Passwort „Superman“ zugeordnet ist. Solche Listen nennt man Regenbogentabellen (rainbow table). Für den Algorithmus MD5 gibt es z.B. unter www.md5online.org ein Tool, das auf solchen Regenbogenlisten basiert und anhand des Hashwertes den Ursprungstext herausfinden kann.

Sicherheit

Um die Passwörter nicht anhand einer Regenbogentabelle zu knacken, sollten keine veralteten Algorithmen wie MD5 oder SHA-1 genutzt werden. Außerdem sollten die Passwörter regelmäßig geändert werden, da auch die Plattformen bei unsicher gewordenen Hashfunktionen auf neue Techniken umstellen und nur bei einem Passwortwechsel genutzt werden können.

Aber auch bei den als „sicher“ geltenden Hashfunktionen kann keine hundertprozentige Sicherheit erwartet werden. Zum einen kann ein manipulierter Hashwert als Prüfsumme an eine Nachricht oder im Internet veröffentlicht werden. Zum anderen ist es nur eine Frage der Zeit, wann die „sicheren“ kryptografischen Hashfunktionen geknackt werden. Denn der Erfolg eines Angriffes ist auch bei den „sicheren“ Hashfunktionen nicht auszuschließen und hängt eher vom Aufwand ab, der sich mit der steigenden Rechenleistung relativieren wird. Ein erfolgreicher Angriff ist derzeit zwar unwahrscheinlich, aber nicht unmöglich. Deswegen sollte man immer auf dem neusten Stand der Technik sein und die Entwicklungen im Auge behalten.

Haben Sie Themen- oder Verbesserungsvorschläge? Kontaktieren Sie uns anonym hier.