Codes und Datenintegrität

Codes, Prüfsummen und kryptographische Hashe

Tom Gries

Dokumenten URL: http://docs.tx7.de/TT-EBD

Autor: Tom Gries <TT-EBD@tx7.de>
@tomo@chaos.social

Version: 7.2.0 vom 02.02.2024

Codes

Unter Code versteht man ein System von Regeln und Übereinkünften, das die Zuordnung von Zeichen oder Zeichenfolgen zweier verschiedener Zeichenvorräte erlaubt. Bekannte Beispiele sind unter anderem der IATA-Flughafencode für einen Flugplatz oder Metropolitan Area. Zum Beispiel BER für alle Flughäfen in Berlin oder FRA für den Flughafen Frankfurt am Main.

Im täglichen Leben kommen wir zum Beispiel mit Barcodes (auf den Lebensmittelpackungen) oder QR-Codes in Berührung.

Prüfsummen

Eine Prüfsumme dient zur Prüfung der Integrität von Daten. Im einfachsten Fall handelt es sich um eine Prüfziffer (wie beim Personalausweis) oder dem Paritätsbit. Einfachere Verfahren erkennen keine "Zahlendreher". Und auch Mehrfachfehler sind im Allgemeinen nicht zu erkennen - besonders nicht, wenn sich diese "aufheben". Beispielsweise bei zwei gekippten Bits.

Komplexere Verfahren sind hingegen so gut, dass sie nicht nur Fehler erkennen, sondern diese auch korrigieren können. Beispiele hierfür sind CRC und der QR-Code.

Fehlerkorrektur in QR-Codes

QR-Codes mit integrierten Texten oder Bildern nutzen das Fehlerkorrektur-verfahren auf Kosten leichter Anfälligkeit einfach aus.

Bei schlechten oder ungünstigen Lichtverhältnissen oder abgenutzten QR-Codes erhöht sich das Nichter-kennungsrisiko.

Prüfsummen in der Praxis

IBAN

Länge	Struktur				Richtlinie
34	Land nach ISO	PZ	BBAN[1]		ISO 13 616
			IID[2]	BAN[3]	
	$l_1 l_2$	pp	x_1 - x_n	y_1 - y_m	

[1]: BBAN - Basic Bank Account Number
n + m <= 30
[2]: IID - Institute Identifikation (BLZ/Bankleitzahl)
[3]: BAN - Bank Account Number (Kontonummer)

Personalausweis

Länge	Struktur																					
36	Seriennummer									PZ	N	Geburtsdatum (invers)						PZ	Ablaufdatum (invers)						PZ
	Behörden-kennzahl				Fortlaufende Ausweisnummer																				
	x_1	x_2	x_3	x_4	y_1	y_2	y_3	y_4	y_5	p_{SN}	D	<	<	j	j	m	m	t	t	p_{GD}	<	j	j	m	m	t	t	p_{AD}

N = Nationalität (D=Deutscher)
p_{SN}: Prüfziffer der Serienummer, p_{GD}: Prüfziffer des Geburtsdatums
p_{AD}: Prüfziffer des Ablaufdatums, p_{ges}: Prüfziffer über alle Ziffern

Prüfsummen in der Kryptografie

Für kryptografische Anwendungen müssen die Hashverfahren "stabiler" (im Sinne von Kollisionsresistenter) sein. Dort reichen einfache Prüfsummen nicht aus. Eine typische Anwendung ist die digitale Signatur.

Auch für Passwörter werden Hashe verwendet. Ziel in modernen Hashverfahren ist es, die Zeit zur Generierung des Hashes so lange wie möglich zu gestalten, ohne dass der User Einschränkungen hat. Dies verlängert die Crackzeit bei Angriffe deutlich. Zusätzlich verfügen sie über ein sogenanntes SALT (Salz), um Doubletten bei gleichen Klartext-Passwörtern zu vermeiden. PEPPER wird auf Servern als weitere Sicherheitsmaßnahme eingesetzt.

Hashes - Beispiele Passwort-Hashe

Verfahren	Erscheinungsjahr	Bemerkung
DES	1975	Salt (2 Zeichen), nur 8 Zeichen Klartext, resistent gegen differenzielle Kryptoanalyse (1991 veröffentlicht).
LM-Hash (LAN Manager)	1988/1989	Kein Salt
MD5	1991	Salt, Geschwindigkeitsoptimiert
NTLM Hash	1993	Basiert auf DES, kein Salt
Bcrypt	1999	Salt, Kostenfaktor zur Geschwindigkeitsreduzierung
Scrypt	2010	Salt, Kostenfaktor zur Geschwindigkeitsreduzierung
Argon2	2014	Salt, Kostenfaktor zur Geschwindigkeitsreduzierung

Anmerkungen oder Fragen?