Multiplikative Hashfunktion

Schlüssel $s \in \{0, 1, \ldots, 2^w - 1\}$. w = Wortlänge

zufälliger Parameter

$z \in \{1, 3, 5, \ldots, 2^w - 1\}$ ungerade

$$h(s) = \lfloor (zs \bmod 2^w) / 2^{w-d} \rfloor$$

$$= (zs \bmod 2^w) >> (w-d) \in \{0, \ldots, \underbrace{2^d - 1}_{=L}\}$$

Beispiel.

w = 32 bits

2^w (4294967296)	10000000000000000000000000000000
z (4102541685)	11110100100001111101000101110101
s (42)	00000000000000000000000000101010
$z \cdot s$	101000000111100100100001011010011001 0
$(z \cdot s) \bmod 2^w$	00011110010010000101110100110010
$((z \cdot s) \bmod 2^w) \operatorname{div} 2^{w-d}$	00011110

d = 8 Bits

Satz 1. $s \neq s'$ fest

$$\Pr[h(s) = h(s')] \leq \frac{2}{2^d}$$

↑ zufälliges z

Beweis:

Lemma 1: $q \in U = \{1, 3, 5, \ldots, 2^w - 1\}$
$f(z) := z \cdot q \bmod 2^w$
ist eine bijektive Abbildung $f: U \to U$

Beweis:

f injektiv: Beweis durch Widerspruch, $z < z'$

$$z \cdot q \equiv z' \cdot q \pmod{2^w}$$
$$(z' - z) \cdot q \equiv 0 \pmod{2^w}$$

$2^w \mid (z' - z) \cdot q$

$(\underbrace{\ldots\ldots 10000}_{w}) \cdot (\underbrace{\ldots\ldots\ldots 1}_{w})$

a Nullen $0 \le a < w$

$= (\ldots\ldots\ldots\ldots\ldots 10000)$

a Widerspruch.

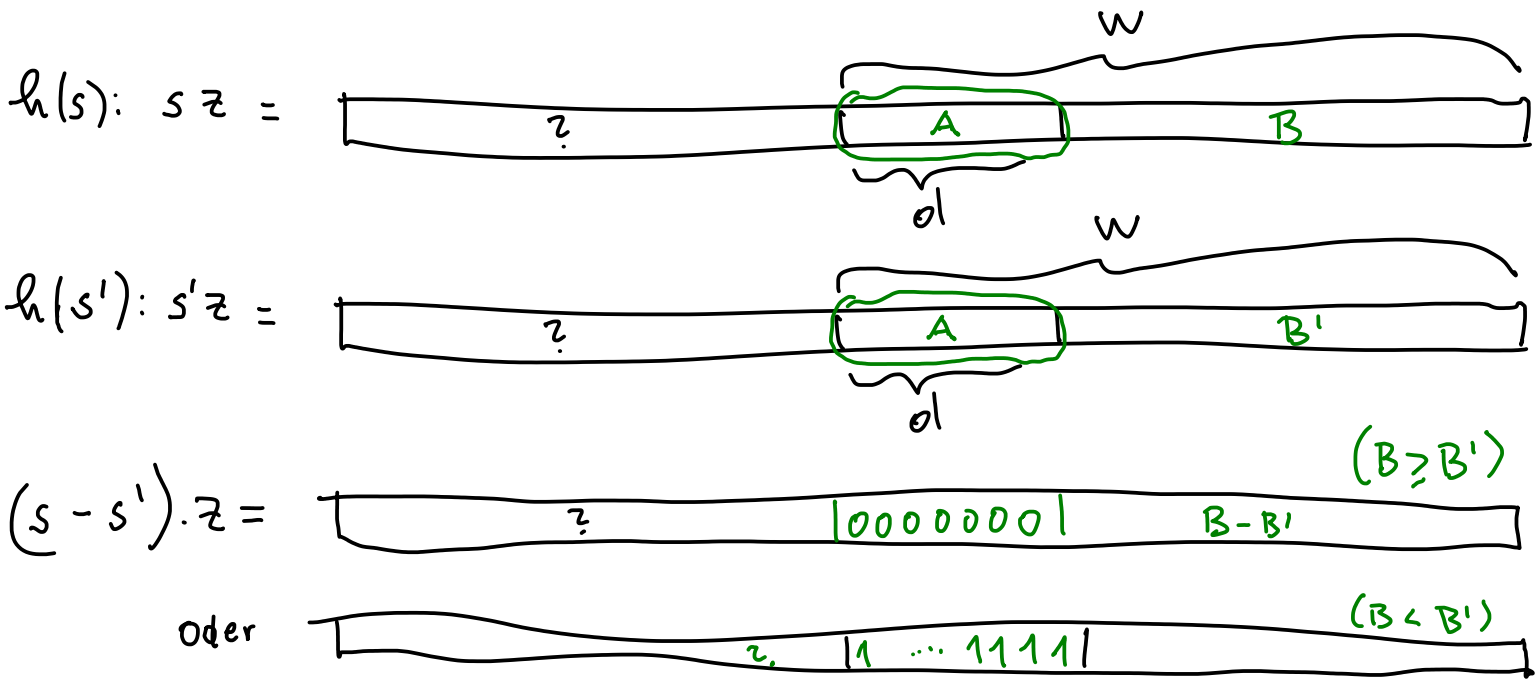
Andere Formulierung: q fest, $z \in U$ zufällig

$z \cdot q \bmod 2^w$

$\underbrace{* - - - - - - - 1}_{w-1 \text{ Zufallsbits}}$

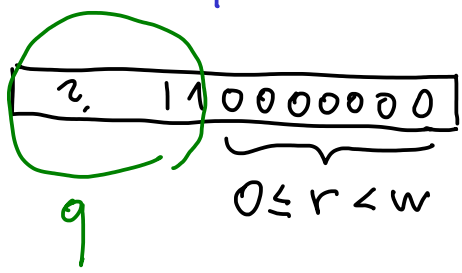
Beweis des Satzes:

$$\Pr\left[h(s) = h(s')\right] \qquad s > s'$$

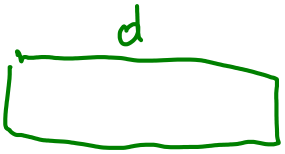


$h(s)=h(s') \Rightarrow (s-s')z \bmod 2^w$ hat an den führenden d Stellen lauter Nullen oder lauter Einsen.

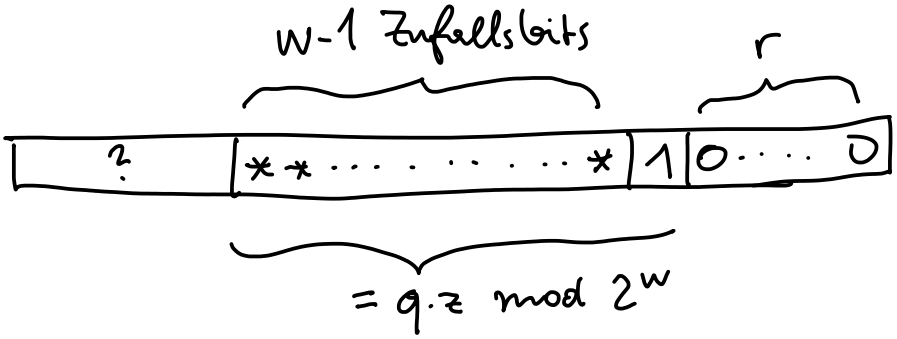
$(s-s')z = q \cdot 2^r$

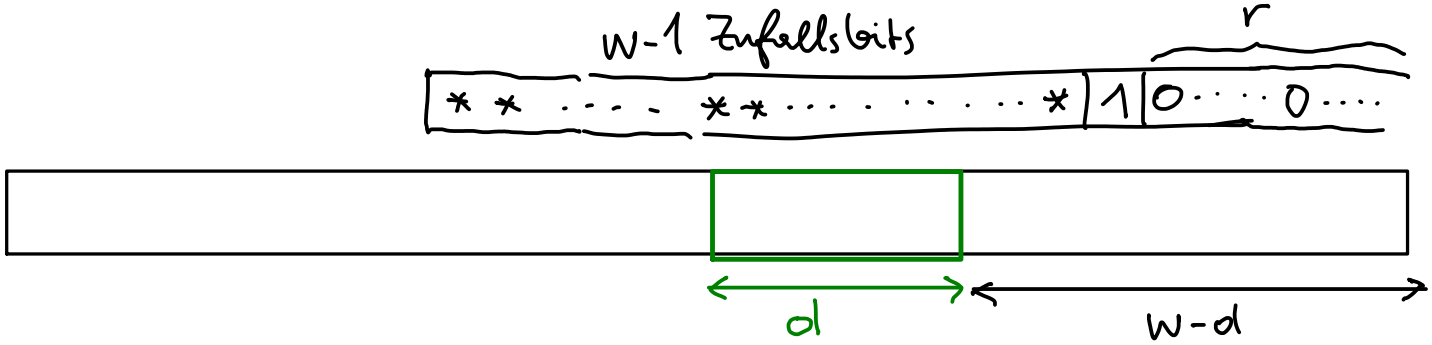


$= q \cdot 2^r$



$(s-s').z = 2^r \cdot q \cdot z =$





Fall 1: $r > w-d$: $Pr[0000..0 \text{ oder } 11..1] = 0$

Fall 2: $r = w-d$: $Pr[0000..0 \text{ oder } 11..1] = \frac{1}{2^{d-1}}$

Fall 3: $r < w-d$: $Pr[0000..0 \text{ oder } 11..1] = \frac{2}{2^d}$

$\leq \frac{2}{2^d}$ □

Andere multiplikative Hashfunktion mit Primzahlen

$L = p$ .. eine Primzahl ... $z \in \{0, \dots, p-1\}$ zufällig

$$h(s) = s \cdot z \mod p$$

Satz 1^p: $s \neq s'$ fest $\in \{0, \dots, p-1\}$

$$Pr[h(s) = h(s')] \leq \frac{1}{p}$$

↑ zufälliges z

Lemma 1^p: $q \in U = \{0, 1, \dots, p-1\}$, $q \neq 0$

$f(z) := z \cdot q \mod p$

ist eine bijektive Abbildung $f: U \to U$

Hashcodes für längere Objekte

$(s_1, s_2, \ldots, s_k)$ k Stücke zu je d Bits $0 \le s_i < 2^d = L$

$z_1, z_2, \ldots, z_k$ k zufällige Multiplikatoren mit je d Bits

z ungerade mit 2d Bits, zufällig.

---

1. $t = s_1 z_1 + \cdots + s_k z_k \mod 2^{2d}$
2. $(w = 2d) \quad \lfloor (t z \bmod 2^{2d}) / 2^d \rfloor = u = h(s_1, \ldots, s_k)$

Satz 2. $(s_1, s_2, \ldots, s_k) \neq (s_1', s_2', \ldots, s_k')$ fest.

$$\Pr[u = u'] \le \frac{3}{2^d}$$

↑ zufälliges $z, z_1, \ldots, z_k$

BEWEIS: o.B.d.A. $s_1 < s_1'$

(1.) $\Pr[t = t'] = \Pr[\underbrace{(s_1' - s_1)}_{>0} z_1 + \underbrace{(s_2' - s_2) z_2 + \ldots + (s_k' - s_k) z_k}_{-A} \equiv 0 \pmod{2^{2d}}]$

$$A := \left(-\left[(s_2' - s_2) z_2 + \ldots + (s_k' - s_k) z_k\right]\right) \bmod 2^{2d}$$

$$\Pr[t - t'] = \Pr[\underbrace{(s_1' - s_1)}_{d \text{ Bits}} \underbrace{z_1}_{d \text{ Bits}} \equiv A \pmod{2^{2d}}]$$

$>0 \quad \ge 0$

$$= \Pr[(s_1' - s_1) z_1 = A] = \Pr\left[z_1 = \frac{A}{s_1' - s_1}\right] \le \frac{1}{2^d}$$

(2.) $\Pr[u = u' \mid t \neq t'] \le \frac{2}{2^d}$ nach Satz 1.

$$\Pr[u = u'] \le \Pr[① \vee ②] \le \frac{1}{2^d} + \frac{2}{2^d} = \frac{3}{2^d}$$

□

JAVA: hashCode() vom Typ int (32 Bit)

hängt zusammen mit equals()

Vertrag: a.equals(b) ⇒ a.hashCode() == b.hashCode()

Kryptographische Hashfunktionen

Aus y lässt sich x mit h(x) = y nicht effizient berechnen

SHA256 Linux: sha256sum