Sommersemester 2005

Hasso- Plattner- Institut, Universität Potsdam

Übung zur Vorlesung

Fachgebiet Internet- Technologien und - Systeme

Meinel / Kutzner

Aufgabenblatt 10
(NUR FÜR ULI-STUDENTEN)

(URL: http://www.hpi.uni- potsdam.de/index.php?id=411)

Abgabe: Mo, 27.06.2005, bis 12 Uhr MEZ (per E-Mail an mathias.kutzner@hpi.uni- potsdam.de)

Thema: Kryptographische Hash- Funktionen

Erreichbare Punkte: 18

Aufgabe 1: 6 Punkte

Machen Sie sich mit dem Algorithmus SHA- 1 vertraut, der zur Erzeugung von Hashwerten verwendet wird. Erklären Sie mit eigenen Worten, wie bei diesem Algorithmus ein letzter unvollständiger Block aufgefüllt wird (Padding).

Füllen Sie den Block 1a7fd53b4c entsprechend auf und geben Sie Ihr Ergebnis an.

Aufgabe 2: 12 Punkte

Schreiben Sie ein Programm, dass SHA- 1 implementiert. Dokumentieren Sie Ihren Quellcode ausführlich. Als Ergebnis soll Ihr Programm zusätzlich zu dem Hashwert tabellarisch W_0 bis W_{79} ausgeben, sowie für jeden Durchlauf *t* die Register A, B, C, D und E.

Ermitteln Sie mit Ihrem Programm den Hashwert für den Block aus Aufgabe 1. Geben Sie sowohl den Quellcode als auch die komplette Ausgabe des Programms ab.

Hinweise:

Zur Berechnung des Hashwertes sind fünf 32- Bit Words notwendig, die folgendermaßen initialisiert werden:

$H_0 = 67452301 \quad H_1 = \text{EFCDAB89} \quad H_2 = 98\text{BADCFE} \quad H_3 = 10325476 \quad H_4 = \text{C3D2E1F0}$

Abhängig vom jeweiligen Verarbeitungsschritt verwendet SHA- 1 eine von vier verschiedenen Grundoperationen. Diese Operationen werden durch die folgende Funktion beschrieben:

$$f_t : \{0,1\}^{32} \times \{0,1\}^{32} \times \{0,1\}^{32} \rightarrow \{0,1\}^{32}$$

$$f_t(B,C,D) = \begin{cases} 5a827999 + ((B \wedge C) \vee (\neg(B) \wedge D)) & 0 \le t \le 19 \\ 6ed9eba1 + (B \oplus C \oplus D) & 20 \le t \le 39 \\ 8f1bbcdc + ((B \wedge C) \vee (B \wedge D) \vee (C \wedge D)) & 40 \le t \le 59 \\ ca62c1d6 + (B \oplus C \oplus D) & 60 \le t \le 79 \end{cases}$$

Bitte wenden!

Die Nachricht wird in die Blöcke M_0 bis M_{n-1} der Länge 512 Bit aufgeteilt (evtl. letzten Block auffüllen). Die Verarbeitung aller 512- Bit Blöcke erfolgt sequentiell. Jeder 512- Bit Block M_i wird in 16 32- Bit Words zerlegt. Aus diesen 16 32- Bit Words werden 80 32- Bit Words W_0 bis W_{79} generiert. Die Erzeugung der 80 Words W_0 bis W_{79} für ein M_i erfolgt folgendermaßen:

- Zerlege M_i in 16 Words und bilde damit W_0 bis W_{15}, wobei W_0 das Wort ganz links ist
- Für $t = 16,\ldots,79$ setze $W_t = S^1(W_{t-16} \oplus W_{t-14} \oplus W_{t-8} \oplus W_{t-3})$
- Setze $A = H_0, B = H_1, C = H_2, D = H_3$ und $E = H_4$
- Für $t = 0,\ldots,79$ setze

$$TEMP = S^5(A) + f_t(B,C,D) + E + W_t$$
$$E = D; D = C; C = S^{30}(B); B = A; A = TEMP$$

- Setze $H_0 = H_0 + A, H_1 = H_1 + B, H_2 = H_2 + C, H_3 = H_3 + D$ und $H_4 = H_4 + E$

Die Funktion S^i führt einen zirkulären Linksshift um i Bits aus.

Nach Verarbeitung des letzten 512- Bit Blocks M_{n-1} ergibt sich der 160- Bit Hashwert durch Aneinanderreihung von H_0 bis H_4.