Universität Duisburg-Essen **SS 2019**
Ingenieurwissenschaften / Informatik **20. Mai 2019**
Dozent: Prof. Dr. Barbara König **Übungsblatt 3**
Übungsleitung: Richard Eggert **Abgabe: 29. Mai 2019**

Formale Aspekte der Softwaresicherheit und Kryptographie

Hinweis: Dies ist ein Testatblatt! Geben Sie Ihre Lösungen bis spätestens Mittwoch, den 29. Mai in der Übung ab! Falls Sie Ihre Lösungen für die Übung schon korrigiert haben wollen, reichen Sie Ihre Lösungen im Raum LF 263 bis Dienstag, 28. Mai, 16:00 ein.
Dieses Übungsblatt wird am 29.05 besprochen.

Aufgabe 10 Miller-Rabin-Primzahltest (4 Punkte)

(a) 15 ist eine zusammengesetzte Zahl. Ermitteln Sie alle ihre Fermat-Zeugen.

(b) Wir betrachten die Carmichael-Zahl $n = 561$. Führen Sie den Primzahltest entsprechend dem Algorithmus auf den Folien für $a = 2$, $a = 9$ und $a = 101$ durch.

Aufgabe 11 Micali-Blum-Pseudo-Zufallsgenerator (4 Punkte)

Folgender Pseudo-Zufallsgenerator - basierend auf der diskreten Exponentialfunktion und dem diskreten Logarithmus – wurde von Micali und Blum vorgeschlagen:

- Wähle eine Primzahl p und eine Primitivwurzel g modulo p.
- Wähle eine Zufallszahl $x \in \{1, \ldots, p-1\}$ und setze $x_1 = x$. Bestimme $x_i = g^{x_{i-1}} \mod p$ für $i = 2, \ldots, k$.
- Setze $b_i = most(x_i)$, wobei b_i das höchstwertige Bit von x_i ist. Genauer:

$$most(x_i) = \begin{cases} 0 & \text{falls } x_i \leq \frac{p-1}{2} \\ 1 & \text{sonst} \end{cases}$$

- Gib die Sequenz $b_k, b_{k-1}, \ldots, b_2, b_1$ als Pseudo-Zufallszahl aus.

(a) Gegeben seien $p = 19$, $g = 3$ und $x = 4$. Bestimmen Sie die dazugehörige Zufallsbitfolge der Länge 5.

(b) Nach welcher Länge fängt eine Zufallsbitfolge spätestens an, sich zu wiederholen?

(c) Ein wichtiges Kriterium für die Güte von Pseudo-Zufallsgeneratoren ist der sogenannte *Nächstes-Bit-Test*. Angenommen wir haben einen probabilistischen Tester, der – gegeben eine Pseudo-Zufallsfolge der Länge $k-1$ – mit einer Wahrscheinlichkeit von $p > \frac{1}{2}$ das nächste Bit ermitteln kann. Beschreiben Sie, wie man damit mit einer gewissen Wahrscheinlichkeit $most(y)$ ermitteln kann, wenn $g^y \mod p$ gegeben ist.

Anmerkung: Falls der diskrete Logarithmus eine Einwegfunktion ist, so ist bekannt, dass jedes probabilistische Polynomzeitverfahren $most(y)$ höchstens mit Wahrscheinlichkeit $\frac{1}{2}$ ermitteln kann, wenn $g^y \mod p$ gegeben ist. D.h., die Existenz eines erfolgreichen Testers würde zeigen, dass *dexp* keine Einwegfunktion ist.

Aufgabe 12 Angriffe auf das RSA-Signaturschema (4 Punkte)

Wir betrachten digitale Signaturen mit Hilfe von RSA, wie sie in der Vorlesung beschrieben wurden. Gegeben sei ein öffentlicher Schlüssel (e, n), der dazugehörige private Schlüssel (d, n) ist unbekannt.

(a) Beschreiben Sie, wie ein Angreifer auf einfache Weise eine Nachricht und eine dazugehörige Signatur erzeugen kann.

Bemerkung: Der Angreifer muss hier nicht in der Lage sein, jede beliebige Nachricht zu signieren, aber er sollte bestimmte Paare von Nachricht und dazugehöriger korrekter Signatur ermitteln können.

(b) Beschreiben Sie, wie ein Angreifer in der Lage ist, eine gegebene Nachricht m korrekt zu signieren, wenn Alice für ihn beliebige Nachrichten $m' \neq m$ signiert.

Bemerkung: Das ist in der Praxis durchaus möglich, da der Angreifer die Nachrichten m' wie Zufallsstrings aussehen lassen kann. Alice signiert dann einfach nur einen Nonce.

Aufgabe 13 Kollisionen und das Geburtstagsproblem (3 Punkte)

Bei Hashfunktionen muss darauf geachtet werden, dass Kollisionen schwer zu finden sind. Das Auftreten von Kollisionen ist verwandt mit dem sogenannten *Geburtstagsproblem:*

> In einem Raum befinden sich n Personen. Wie groß ist die Wahrscheinlichkeit, dass zwei davon am gleichen Tag Geburtstag haben?

Wie groß muss n sein, damit diese Wahrscheinlichkeit größer als $\frac{1}{2}$ ist?