

Übungen zur Vorlesung „Betriebssysteme“

Blatt 13

Aufgabe 1

Beim RSA-Verschlüsselungsverfahren werden der öffentliche Schlüssel (*public key*) und der geheime Schlüssel (*secret key*) nach dem folgenden Algorithmus bestimmt.

1. Wähle große (> 100 Stellen) Primzahlen p, q .
 2. $n := p \cdot q$.
 3. Wähle $e < n$ mit $\text{ggT}(e, \varphi(n)) = 1$. (* $\varphi(n) = (p - 1)(q - 1)$ *)
 4. Berechne $d \in \mathbf{N}$ mit $e \cdot d = 1 \pmod{\varphi(n)}$. (* Erweiterter Euklidischer Algorithmus *)
 5. public key := (e, n) ; secret key := (d, n) .
- a) Führen Sie die Berechnung eines öffentlichen und eines entsprechenden geheimen Schlüssels für die Wahl $p = 127, q = 139$ von Hand durch.
- b) Verschlüsseln Sie eine beliebige (Teil-)Nachricht, und prüfen Sie durch Entschlüsselung Ihres chiffrierten Textes, ob Sie korrekt gearbeitet haben.

Aufgabe 2

Der Aufwand zur Entschlüsselung einer mit dem RSA-Verfahren chiffrierten Nachricht ohne Kenntnis des geheimen Schlüssels entspricht dem Aufwand zur Faktorisierung natürlicher Zahlen. Eine naheliegende Idee besteht darin, die zu faktorisierende Zahl M nacheinander durch die aufeinanderfolgenden Primzahlen $2, 3, 5, 7, 11, \dots$ zu dividieren bis die kleinste Primzahl p gefunden wird, die Teiler von M ist. Dann wird dieses Verfahren rekursiv auf M/p angewendet. Somit erhält man nacheinander die Primfaktoren $p_1 \leq p_2 \leq \dots \leq p_t$.

- a) Formulieren Sie einen Algorithmus, der diese Idee umsetzt. Achten Sie dabei insbesondere auf geeignete Abbruchbedingungen.
- b) Führen Sie Ihren Algorithmus von Hand für die Zahlen 510510, 425253, 99983 durch.
- c)* Wieviele Divisionen benötigt der Algorithmus? Geben Sie alternative (effizientere) Faktorisierungsalgorithmen an.

Aufgabe 3

Nennen Sie ein einfaches Beispiel für eine mathematische Funktion, die in erster Näherung eine Einwegfunktion ist.

Aufgabe 4

Das Morris-Thompson-Schutzverfahren mit den n-Bit-Zufallszahlen (Salt) wurde entwickelt, um es einem Eindringling zu erschweren, eine große Anzahl von Paßwörtern aufzudecken, indem er vorab geläufige Zeichenketten verschlüsselt. Schützt dieses Verfahren auch vor einem Studenten, der versucht, das Superuser-Passwort auf seiner Maschine zu erraten? Nehmen Sie an, daß die Paßwortdatei zum Lesen freigegeben ist.

Aufgabe 5

Funktioniert ein Angriff mit Trojanischen Pferden in einem System, das mit Capabilities geschützt ist?

Aufgabe 6

Nennen Sie eine Eigenschaft des C-Compilers, die eine große Anzahl von Sicherheitslücken eliminieren könnte. Wieso wird diese nicht häufiger implementiert?

Aufgabe 7

Wie kann ein parasitäres Virus sicherstellen, daß (a) es vor dem Wirtsprogramm ausgeführt wird und (b) die Kontrolle zu seinem Wirt zurückgeben kann, nachdem es erledigt hat, was auch immer es getan hat.

Aufgabe 8

Modifizieren Sie die ACL für eine Datei derart, daß ein Zugriff gewährt bzw. verboten wird, der nicht mittels des rwx-Systems in UNIX beschrieben werden kann. Erklären Sie diese Modifikation.

Aufgabe 9

Capabilities und Zugriffskontrolllisten sind zwei unterschiedliche Schutzmechanismen, die wir diskutiert haben. Geben Sie an, welcher dieser Mechanismen für jede der folgenden Problemstellungen genutzt werden kann:

- (a) Ken will, daß seine Dateien von jedermann außer seinem Bürokollegen gelesen werden kann.
- (b) Mitch und Steve wollen einige geheime Dateien gemeinsam nutzen.
- (c) Linda will, daß einige ihrer Dateien öffentlich zugänglich sind.