Universität Dortmund
Lehrstuhl Verteilte Systeme

Dr. Michael Zapf

Michael.Zapf@cs.uni-dortmund.de
Michael.Zapf@uni-kassel.de

tdi@ls4.cs.uni-dortmund.de

Übung zur Vorlesung „Techniken und Dienste des Internets“ - SS 2007

Blatt 6

Ausgabe 16.05. – Abgabe 23.05.

Aufgabe 6.1 (10 Punkte)

Wenn ein DNS-Server eine Anfrage nicht selbst beantworten kann, kann er die Anfrage an andere Server delegieren. Die Antworten werden dann für eine gewisse Zeit im Cache gespeichert und an den ursprünglichen Anfrager übermittelt. DNS arbeitet auf Basis von UDP. Warum kann dies zu einer gefährlichen Sicherheitslücke werden?

Aufgabe 6.2 (10 Punkte)

Das RSA-Verfahren basiert darauf, dass große Primzahlprodukte p*q nicht effizient in ihre Faktoren p und q zerlegt werden können. Aber wie stellt man fest, ob p und q prim sind, wenn diese immer noch Hunderte von Dezimalstellen aufweisen? Recherchieren Sie im Internet nach geeigneten Verfahren und geben Sie das Prinzip kurz wieder.

Aufgabe 6.3 (5 Punkte)

Angenommen, Alice bemerkt, dass ihr privater RSA-Schlüssel (d1, n1) genau gleich dem öffentlichen RSA-Schlüssel (e2, n2) von Bob ist, also d1=e2 und n1=n2. Sollte Alice sich schleunigst ein neues Schlüsselpaar generieren? Warum? (Hinweis: Überlegen Sie, warum RSA so sicher ist.)

Aufgabe 6.4 (15 Punkte)

Alice hat sich einen ganz einfachen Message-Digest-Algorithmus ausgedacht, um ihre Nachrichten zu sichern: Ihre Nachricht besteht aus einer Folge von Bytes, also zählt sie alle Bytes zusammen und behält die niederwertigsten 8 Bit (d.h. sie rechnet b[0] + b[1] + ... + b[n] mod 256). Diese Summe verschlüsselt sie mit ihrem privaten Schlüssel. Ihr öffentlicher Schlüssel ist frei verfügbar.

a) Zeigen Sie, dass Sie nach bereits 20 Nachrichten von Alice mit mehr als 50% Wahrscheinlichkeit mindestens zwei Nachrichten mit gleichem Abdruck (Digest) erhalten haben.

b) Wie viele Nachrichten benötigen Sie bei einem Nachrichtenabdruck der Länge n Bit (für größere n), bevor mit mehr als 50% Wahrscheinlichkeit eine Kollision eintritt? Erreicht sie irgendwann 100%?

c) Alice lässt sich die Nachrichten von einer (vermeintlich) vertrauenswürdigen Person Carol schreiben und verschicken, d.h. diese Person hat die Möglichkeit, Alice einen Text vorzulegen, der inhaltlich zwar das Gewünschte aussagt, aber bezüglich der Wortwahl frei ist. Carol möchte Alice diskreditieren. Können Sie sich einen Angriff vorstellen, der diese Hashkollision ausnutzt?

d) Was passiert, wenn Alice künftig ihre Briefe nur noch selbst schreibt? Ist der Angriff aus c) noch möglich, und welchen Aufwand muss Carol dann treiben?