4 Science Challenge, 23. Wettbewerb

Aufgabe 3: Kryptographie

Die Kryptographie – die Wissenschaft der Verschlüsselung – ist seit Jahrtausenden ein integraler Bestandteil der Kommunikation und des Informationsaustauschs. Sie ermöglicht es uns, Botschaften sicher zu übertragen, und schützt unsere digitalen Informationen und Systeme. Dieses Aufgabenblatt führt euch in einige faszinierende Aspekte der Kryptographie ein. Ihr werdet aufgefordert, eure mathematischen und logischen Fähigkeiten einzusetzen, um verschiedene kryptographische Probleme zu lösen.

Aufgabe 1 (6 Punkte): Modulare Arithmetik

Die modulare Arithmetik, oft einfach „Modulo" genannt, bezieht sich auf den Rest, der nach der Division einer Zahl durch eine andere übrig bleibt. In der ersten Aufgabe des Wettbewerbs habt ihr diese Art des Rechnens bereits kennengelernt. Man sagt, dass zwei Zahlen a und b „kongruent modulo n" sind, wenn sie bei Division durch n denselben Rest ergeben. Man schreibt auch $a \equiv b \bmod n$. In der Kryptographie wird die modulare Arithmetik häufig verwendet, insbesondere in Protokollen wie RSA oder Diffie-Hellman.

a) Berechnet:
1.) $15 \bmod 7$
2.) $124 \bmod 9$
3.) $64 \bmod 6$

b) Die üblichen Potenzgesetze lassen sich auch für das modulare Potenzieren anwenden:

I) $x^{a+b} \bmod d \equiv ((x^a \bmod d) \cdot (x^b \bmod d)) \bmod d$

II) $x^{a \cdot b} \bmod d \equiv (x^a \bmod d)^b \bmod d$

Berechnet damit:
$23115^{23} \bmod 2$

c) Ein quadratischer Rest modulo n ist eine Zahl, die kongruent zu einem Quadrat einer natürlichen Zahl modulo n ist. Zum Beispiel sind 0 und 1 die einzigen quadratischen Reste modulo 4.

Bestimmt alle quadratischen Reste modulo 7.

Aufgabe 2 (3 Punkte): Diskreter Logarithmus

Der diskrete Logarithmus ist ein schwer zu lösendes Problem, das die Basis der Sicherheit von vielen kryptographischen Algorithmen bildet. Angenommen, wir haben eine Zahl g (genannt Generator), eine Primzahl p und eine Zahl h. Das Problem besteht darin, den Exponenten x zu finden, für den $g^x \equiv h \bmod p$ gilt.

Gegeben: $g = 3$, $p = 17$ und $h = 15$

Findet den Wert von x.

Aufgabe 3 (3 Punkte): Diffie-Hellman-Schlüsselaustausch

Der Diffie-Hellman-Schlüsselaustausch ermöglicht es zwei Parteien, über einen unsicheren Kanal einen gemeinsamen geheimen Schlüssel zu erzeugen. Sie verwenden dazu eine öffentliche Basis g und einen Modulus p. Jede Partei wählt einen privaten Schlüssel (Alice a, Bob b) und berechnet einen öffentlichen Schlüssel ($A \equiv g^a \bmod p$ für Alice, $B \equiv g^b \bmod p$ für Bob). Sie tauschen diese öffentlichen Schlüssel aus. Der gemeinsame geheime Schlüssel wird als $(g^b)^a \bmod p \equiv (g^a)^b \bmod p$ berechnet, was $g^{ab} \bmod p$ ergibt. Durch das diskrete Logarithmusproblem bleibt dieser Schlüssel geheim.

a) Gegeben:
- Öffentliche Basis $g = 5$
- Öffentlicher Modulus $p = 23$
- Alices privater Schlüssel $a = 6$

Berechnet Alices öffentlichen Schlüssel.

b) Gegeben:
- Öffentliche Basis $g = 5$
- Öffentlicher Modulus $p = 23$
- Alices öffentlicher Schlüssel $A = 8$
- Bobs privater Schlüssel $b = 7$

Berechnet Bobs öffentlichen Schlüssel und den gemeinsamen geheimen Schlüssel.

Aufgabe 4 (8 Punkte): XOR-Verschlüsselung

XOR ist eine binäre Operation, die zwei Bits vergleicht. Wenn die Bits gleich sind, gibt sie 0 zurück, wenn sie unterschiedlich sind, gibt sie 1 zurück. In der Kryptographie wird XOR häufig verwendet, um einen Klartext mit einem Schlüssel zu „verknüpfen".

a) Gegeben ist der folgende Klartext in Binärform: 110101 und der Schlüssel: 001011.

Was ist der verschlüsselte Text?

b) Ein Nachrichtendienst hat den folgenden verschlüsselten Text in Binärform erhalten:
101010101001010110011001100110011010011010011011110

Es ist bekannt, dass die Originalnachricht das Wort "SECRET" enthält, das in eine Binärform umgewandelt wurde (Hinweis: ASCII). Eure Aufgabe ist es, den verwendeten Schlüssel zu finden.

c) Verwendet den Diffie-Hellman-Schlüsselaustausch für die Generierung eines gemeinsamen geheimen Schlüssels und anschließend XOR-Verschlüsselung für die Nachrichtenübermittlung.
- Öffentliche Werte: Basis $g = 5$ und Modulus $p = 23$
- Alice wählt einen geheimen Wert $a = 6$
- Bob wählt einen geheimen Wert $b = 15$

Alice möchte die Nachricht "MOIN" an Bob senden.
Verwendet den gemeinsamen geheimen Schlüssel als Schlüssel für die XOR-Verschlüsselung.
Konvertiert die Nachricht und den Schlüssel in Binärformat (ASCII-Codierung).
Führt die XOR-Verschlüsselung durch und „sendet" die verschlüsselte Nachricht.
Bob soll die Nachricht entschlüsseln.

Aufgabe 5 (10 Punkte): Verschlüsselung mit RSA

Das RSA-Kryptosystem, benannt nach seinen Erfindern Rivest, Shamir und Adleman, wurde 1977 entwickelt. Es revolutionierte die Kryptographie durch die Einführung einer öffentlichen Schlüsseltechnologie, die das Teilen eines geheimen Schlüssels überflüssig machte. RSA wird weitreichend für sichere Datenübertragungen, insbesondere im Internet, eingesetzt.

Das **RSA-Verfahren** funktioniert in folgenden Schritten:

1. Wählt zwei Primzahlen p und q und berechnet ihr Produkt $n = p \cdot q$. Die Zahl n wird auch als **Modul** bezeichnet.
2. Berechnet $\varphi(n) = (p - 1) \cdot (q - 1)$
3. Wählt eine Zahl e, für die $1 < e < \varphi(n)$ gelten soll, und die teilerfremd zu $\varphi(n)$ ist. Diese Zahl e ist unser **Verschlüsselungsexponent**.
4. Berechnet eine Zahl d, für die gelten soll $d \cdot e \equiv 1 \bmod \varphi(n)$ (Hinweis: Modulo-Rechnung, erweiterter euklidischer Algorithmus). Dieses d wird auch als **Entschlüsselungsexponent** bezeichnet.

Der **öffentliche Schlüssel** (also das, was alle sehen dürfen) besteht dann aus e und n. Nachrichten verschlüsselt man dann, indem man sie in Zahlenfolgen umwandelt und dann für jede Zahl m der Nachricht eine Zahl c mit

$$c \equiv m^e \bmod n$$

berechnet (Hinweis: modulares Potenzieren).

Alle anderen Informationen bleiben geheim. Die Entschlüsselung erfolgt, indem man für die übertragenen Zahlen c den Wert $m \equiv c^d \bmod n$ berechnet und so wieder die ursprüngliche Nachricht erhält.
Auf diese Art kann man einzelne Buchstaben oder auch längere Zeichenketten auf einmal verschlüsseln.

a) Verschlüsselt die Nachricht "Hallo" mit den Parametern $p = 13$, $q = 17$, $e = 11$. Um die Buchstaben in Zahlen umzuwandeln, identifiziert diese einfach mit ihrer Stellung im Alphabet (A=1, B=2, etc.).
b) Entschlüsselt die Nachricht
076 111 056 164 185 041 076 111 184 200 041
mit den Parametern $p = 13$, $q = 17$, $e = 11$.
c) Wo liegen eurer Meinung und Erfahrung nach Vor- und Nachteile des RSA-Verfahrens? Begründet eure Antworten.

Viel Erfolg bei der dritten Aufgabe!

Allgemeine Hinweise
Einsendeschluss: Sonntag, 07. Januar 2024, 19:59 Uhr

Gebt eure Lösungen über Stud.IP ab: https://studip.uni-hannover.de

Das zulässige Dateiformat für die zusammengeschriebene Lösung (mit eingebetteten Bildern) ist PDF. Bitte ladet eure Dateien rechtzeitig hoch.
Gebt innerhalb der Datei euren Teamnamen, die Namen der Teammitglieder sowie deren Schulen an. Benennt eure Datei nach folgendem Schema: „Teamname_Aufgabe3".

Das Hochladen funktioniert wie folgt:
Loggt euch mit den bei eurer Anmeldung zur 4 Science Challenge angelegten Zugangsdaten auf der Stud.IP-Seite ein (bitte nutzt dazu den „Login ohne WebSSO"). Geht dann auf „Meine Veranstaltungen" und auf die 4 Science Challenge 2023/2024. Geht dann oben auf „Dateien" und auf den Ordner „Upload Aufgabe 3". Dort könnt ihr entweder über „Dokument hinzufügen" oder über „Dateien hochladen" eure Lösungsdatei hochladen.

Wenn ihr die Datei hochgeladen habt, öffnet sich ein Fenster, in dem u. a. nach Lizenzinformationen gefragt wird. Dieses braucht ihr nicht weiter zu beachten und könnt einfach auf „Speichern" klicken. Bitte achtet darauf, dass ihr eure Dateien wirklich innerhalb des Ordners „Upload Aufgabe 3" hochladet und nicht außerhalb davon, da ansonsten die anderen Teams eure Dateien sehen können.
Die Teilnahmebedingungen und weitere Informationen findet ihr unter
www.uni-hannover.de/4sciencechallenge

Der Rechtsweg ist ausgeschlossen.