Computersystemsicherheit – Übungsblatt Nr. 2 – Lösung

Marc Fischlin, Jacqueline Brendel, Christian Janson
TU Darmstadt, 09 November 2018

Gruppenübung. Die Übungsaufgaben in diesem Bereich sind Gegenstand der Übungen in der Woche vom 12.11.2018 – 16.11.2018.

Aufgabe 1 (Verständnisaufgaben). In dieser Aufgabe prüfen wir unser Verständnis über den Inhalt der Vorlesung.

a) Warum sollte man die Blockchiffe DES nicht mehr benutzen?

Lösung.

Schlüssellänge (56 Bits) ist nach heutigem Stand zu kurz.

b) Welche Schlüssellängen werden bei AES verwendet?

Lösung.

Mögliche Schlüssellängen sind 128 Bits, 192 Bits und 256 Bits.

c) Was besagt die funktionale Korrektheit eines Public-key Verschlüsselungsverfahren?

Lösung.

Für alle Nachrichten m und alle Schlüsselpaare (pk, sk) gilt: $\mathrm{Dec}(sk, \mathrm{Enc}(pk, m)) = m$.

d) Auf welchem schwierigen Problem beruht die Sicherheit des Diffie-Hellman Schlüsselaustauschs?

Lösung.

Diskreter Logarithmus.

e) Erinnern Sie sich an das RSA-Verschlüsselungsverfahren und formalisieren Sie wie die Verschlüsselung und Entschlüsselung funktioniert.

Lösung.

Verschlüsselung: $\mathrm{Enc}(pk, m) = m^e \bmod N$.
Entschlüsselung: $\mathrm{Dec}(sk, c) = c^d \bmod N$.

f) Was ist Hybride Verschlüsselung?

Lösung.

Hybride Verschlüsselung ist eine Kombination aus asymmetrischer und symmetrischer Verschlüsselung. Zum Beispiel wird bei der E-Mail Verschlüsselung hybride Verschlüsselung genutzt. Ein zufällig generierter symmetrischer Schlüssel verschlüsselt die E-Mail. Der Schlüssel wird dann mit asymmetrischer Verschlüsselung übertragen. Diese Kombination führt zu einem schnellen Transfer von symmetrischen Schlüsseln und vereint die Vorteile

beider Verfahren: Performanz und Schlüsselaustausch.

Aufgabe 2 (Euklidischer Algorithmus). In dieser Aufgabe beschäftigen wir uns mit dem *Euklidischen Algorithmus* und dem *Erweiterten Euklidischen Algorithmus*.

Der Euklidische Algorithmus ist ein bekannter Algorithmus mit welchem sich der *größte gemeinsame Teiler* (ggT) zweier natürlicher Zahlen berechnen lässt. Der Algorithmus basiert auf wiederholter Division mit Rest bis die Sequenz terminiert. Der Euklidische Algorithmus beginnt mit den beiden natürlichen Zahlen a und b für welche der ggT bestimmt werden soll. Formal lässt sich der Algorithmus folgendermaßen beschreiben.

$$\begin{aligned} a &= q_1 b + r_1 \quad \text{mit } 0 \leq r_1 < b \\ b &= q_2 r_1 + r_2 \quad \text{mit } 0 \leq r_2 < r_1 \\ r_1 &= q_3 r_2 + r_3 \quad \text{mit } 0 \leq r_3 < r_2 \\ &\dots \\ r_{n-2} &= q_n r_{n-1} + r_n \quad \text{mit } 0 \leq r_n < r_{n-1} \\ r_{n-1} &= q_{n+1} r_n \end{aligned}$$

Diese Sequenz terminiert, da die Reste immer strikt kleiner werden und der letzte Rest r_n ist der ggT von a und b, also $\mathrm{ggT(a, b)} = \mathrm{r_n}$.

a) Berechnen Sie den ggT für $a = 1337$ und $b = 42$.

Lösung.
Mit dem Euklidischen Algorithmus erhalten wir:

$$\begin{aligned} 1337 &= 31 \cdot 42 + 35 \\ 42 &= 1 \cdot 35 + 7 \\ 35 &= 5 \cdot 7 \end{aligned}$$

Von der obigen Rechnung kann man ablesen, dass $\mathrm{ggT}(1337, 42) = 7$.

Mit einer Erweiterung des obigen Algorithmus lassen sich neben dem ggT für zwei natürliche Zahlen a und b noch zwei ganze Zahlen x und y bestimmen, so dass die Gleichung $c \cdot \mathrm{ggT(a, b)} = \mathrm{x \cdot a + y \cdot b}$ erfüllt ist. Dabei gibt es verschiedene Ansätze dies zu tun. Hier stellen wir kurz die *rekursive* Variante vor.
Erstellen Sie eine Tabelle mit 5 Spalten und die Anzahl der Zeilen hängt vom Euklidischen Algorithmus ab.

a	b	q	x	y

Zuerst berechnet man wie zuvor beschrieben den $\mathrm{ggT(a, b)}$ und trägt die entsprechenden Elemente in die Tabelle:

a	b	q	x	y
a	b	q_1		
b	r_1	q_2		
r_1	r_2	q_3		
$\vdots$	$\vdots$	$\vdots$		
r_{n-2}	r_{n-1}	q_n		

Danach berechnet man rekursiv von unten nach oben die Werte für alle x_i und y_i. Dabei gilt die Vorschrift

$$x_i := y_{i+1} \text{ und } y_i := x_{i+1} - q_i \cdot y_{i+1}$$

für $i \in \{1, \ldots, n\}$ mit $x_{n+1} := 0$ und $y_{n+1} := 1$ und erhält folgende Tabelle:

a	b	q	x	y
a	b	q_1	x_1	y_1
b	r_1	q_2	x_2	y_2
r_1	r_2	q_3	x_3	y_3
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
r_{n-2}	r_{n-1}	q_n	x_n	y_n

Die Werte für x_1 und y_1 liefern die gesuchten Werte x, y, so dass die Gleichung $x \cdot a + y \cdot b = \text{ggt(a, b)}$ erfüllt ist (also $c = 1$). Soll die Gleichung für ein beliebiges Vielfaches des ggT erfüllt sein, müssen die Werte x_1 und y_1 um den Faktor c erweitert werden.
Der Erweiterte Euklidische Algorithmus wird beispielsweise benötigt um im RSA-Verschlüsselungsverfahren das multiplikative Inverse von e, also d, zu bestimmen.

b) Berechnen Sie mit dem Erweiterten Euklidischen Algorithmus $42 \cdot x + 13 \cdot y = 1$ mit $x, y \in \mathbb{Z}$.

Lösung.
Wir stellen zuerst eine Tabelle mit 5 Spalten auf.

a	b	q	x	y

Wir beginnen nun mit dem Euklidischen Algorithmus zu rechnen und erhalten

$$42 = 3 \cdot 13 + 3$$
$$13 = 4 \cdot 3 + 1$$

Wir tragen die resultierenden Parameter in die Tabelle:

a	b	q	x	y
42	13	3		
13	3	4		

Wir berechnen jetzt rekursiv die fehlenden Parameter und erhalten

$$x_2 = y_3 = 1 \text{ und } y_2 = x_3 - q_2 \cdot y_3 = 0 - 4 \cdot 1 = -4$$
$$x_1 = y_2 = -4 \text{ und } y_1 = x_2 - q_1 \cdot y_2 = 1 - 3 \cdot (-4) = 13$$

Damit folgt

a	b	q	x	y
42	13	3	-4	13
13	3	4	1	-4
3	1		0	1

Die Werte x_1 und y_1 sind die gesuchten Werte für die Gleichung. Wir können das Ergebnis auch noch durch die Probe durch Einsetzen verifizieren. Also $42 \cdot (-4) + 13 \cdot 13 = 1$.

c) Berechnen Sie mit dem Erweiterten Euklidischen Algorithmus $483 \cdot x + 136 \cdot y = 3$ mit $x, y \in \mathbb{Z}$.

Lösung.
Die Lösung der Aufgabe folgt analog zur vorherigen Lösung .
Nach der Anwendung des Euklidischen Algorithmus und rekursivem Berechnen der Werte erhalten wir

a	b	q	x	y
483	136	3	-29	103
136	75	1	16	-29
75	61	1	-13	16
61	14	4	3	-13
14	5	2	-1	3
5	4	1	1	-1
4	1		0	1

Die Werte $x_1 = -29$ und $y_1 = 103$ sind Lösungen für die Gleichung $483 \cdot x + 136 \cdot y = 1$.
Wir suchen die Lösung für die Gleichung $483 \cdot x + 136 \cdot y = 3$ und müssen daher die Werte x_1 und y_1 um den Faktor 3 zu erweitern. Also $x = -87$ und $y = 309$.

Aufgabe 3 (Eulersche φ-Funktion). Die Eulersche φ-Funktion ist eine Funktion welche für jede natürliche Zahl n die Anzahl der zu n teilerfremden natürlichen Zahlen bestimmt, die nicht größer als n sind. Formal bedeutet dies

$$\varphi(n) := |\{a \in \mathbb{N} | 1 \leq a \leq n \text{ und } \mathrm{ggT(a, n)} = 1\}|.$$

Es gelten für die Eulersche φ-Funktion folgende Rechenregeln:

- Für $n > 1$ gilt $\varphi(n) = n \prod_{p|n} \left(1 - \frac{1}{p}\right)$
- Für p ist Primzahl gilt $\varphi(p) = p - 1$ und $\varphi(p^n) = p^{n-1}(p-1)$ für alle $n \in \mathbb{N}$.
- Für $n = p \cdot q$ mit Primzahlen $p \neq q$ gilt $\varphi(n) = (p-1)(q-1)$
- Für teilerfremde Elemente $m, n \in \mathbb{Z}$ (d.h. $\mathrm{ggT(m, n)} = 1$) gilt $\varphi(m \cdot n) = \varphi(m) \cdot \varphi(n)$

Berechnen Sie die Eulersche φ-Funktion für die Werte $n = 11, 16, 20, 42, 72, 1337$ mithilfe der obigen Rechenregeln.

Lösung.
$n = 11$: Da 11 prim ist gilt: $\varphi(11) = 11 - 1 = 10$.
$n = 16$: Es gilt: $\varphi(16) = \varphi(2^4) = 2^3 \cdot (2-1) = 8$.
$n = 20$: Es gilt: $\varphi(20) = \varphi(2^2) \cdot \varphi(5) = 2 \cdot 4 = 8$.
$n = 42$: Es gilt: $\varphi(42) = \varphi(2) \cdot \varphi(3) \cdot \varphi(7) = 1 \cdot 2 \cdot 6 = 12$.
$n = 72$: Es gilt: $\varphi(72) = \varphi(2^3 \cdot 3^2) = 72 \cdot (1 - \frac{1}{2}) \cdot (1 - \frac{1}{3}) = 24$.
$n = 1337$: Es gilt: $\varphi(1337) = \varphi(7) \cdot \varphi(191) = 6 \cdot 190 = 1140$.

Aufgabe 4 (RSA). In der Vorlesung haben Sie das RSA-Verschlüsselungsverfahren kennengelernt.

a) Gegeben sei nur der RSA public key mit $pk = (N, e) = (247, 7)$. Verschlüsseln Sie die Nachricht $m = 16$.

Lösung.
Nach Aufgabe 1e) gilt für die Verschlüsselung $\text{Enc}(pk, m) = m^e \bmod N$. Einsetzen liefert $c = \text{Enc}((247, 7), 16) = 16^7 \bmod 247 = 55$.

b) Entschlüsseln Sie die Nachricht $c = 2$.
Hinweis: Ermitteln Sie zunächst den geheimen/privaten Schlüssel, d.h. den Exponenten d, welcher das multiplikative Inverse zu $e \bmod \varphi(N)$ ist. Um diesen Exponenten zu berechnen nutzt man den Zusammenhang zwischen public key und secret key aus: $e \cdot d \equiv 1 \bmod \varphi(N)$.

Lösung.
Um zu entschlüsseln braucht man den geheimen Schlüssel, d.h. den Exponenten d. Dieser lässt sich mit der Kongruenz $e \cdot d \equiv 1 \bmod \varphi(N)$ bestimmen.
Zuerst berechnen wir die Eulersche φ-Funktion von $N = 247$ mit den Formeln aus Aufgabe 2. Dafür zerlegen wir N in die Primzahlen p, q und erhalten $N = p \cdot q = 13 \cdot 19 = 247$. Also folgt $\varphi(247) = (13 - 1) \cdot (19 - 1) = 216$. Nun betrachten wir die Gleichung $7 \cdot d \equiv 1 \bmod 216$.
Mit Hilfe des erweiterten Euklidischen Algorithmus berechnen wir jetzt den Exponenten d welcher das multiplikative Inverse zu $e \bmod \varphi(N)$ ist. Also es gilt $e \cdot d + k \cdot \varphi(N) = 1 = \text{ggT}(e, \varphi(N))$. In unserem konkreten Fall heißt das wir wollen die Gleichung $7 \cdot d + k \cdot 216 = 1$ lösen. Durch Anwenden des Erweiterten Euklidischen Algorithmus erhalten wir

a	b	q	k	d
216	7	30	-1	31
7	6	1	1	-1

Damit erhalten wir $k = -1$ und $d = 31$. Dies kann noch durch die Probe verifiziert werden: $7 \cdot 31 + (-1) \cdot 216 = 1$.
Nun können wir c entschlüsseln und erhalten $m = c^d \bmod (N) = 2^{31} \bmod 247 = 193$.

c) Angenommen $p = 5$ und $q = 3$ werden für die Generierung der folgenden RSA-Schlüssel verwendet:

- $d = 4, e = 2, N = 15$
- $d = 4, e = 3, N = 15$
- $d = 3, e = 3, N = 15$

Welcher dieser Schlüssel ist gültig? Begründen Sie kurz warum.

Lösung.
Dies kann überprüft werden durch einsetzen in die Kongruenz $e \cdot d \equiv 1 \bmod \varphi(N)$.

- Mit $d = 4, e = 2, N = 15$ folgt: $2 \cdot 4 \equiv 0 \bmod 8$ und erfüllt nicht die obige Kongruenz.
- Mit $d = 4, e = 3, N = 15$ folgt: $3 \cdot 4 \equiv 4 \bmod 8$ und erfüllt nicht die obige Kongruenz.
- Mit $d = 3, e = 3, N = 15$ folgt: $3 \cdot 3 \equiv 1 \bmod 8$ und erfüllt somit die obige Kongruenz und ist daher ein gültiges Schlüsselpaar.

Aufgabe 5 (Diffie-Hellman Schlüsselaustausch). In der Vorlesung haben Sie den Schlüsselaustausch nach Diffie und Hellman kennengelernt. Dieser ist dazu gedacht, dass zwei Kommunikationspartner, die miteinander verschlüsselt kommunizieren wollen, durch das Schlüsselaustauschprotokoll am Ende einen gemeinsamen Schlüssel besitzen, der für die folgende Kommunikation verwendet wird.

a) Erinnern Sie sich nochmal wie der Diffie-Hellman Schlüsselaustausch funktioniert. Welche Parameter sind jedem zugänglich? Welche Parameter müssen geheim gehalten werden? Formalisieren Sie, dass beide Kommunikationspartner am Ende den gleichen Schlüssel besitzen.

Lösung.
Öffentliche Parameter sind die Elemente g und p. Alice und Bob müssen ihre gewählten Elemente x und y geheim halten. Alice berechnet $X = g^x \bmod p$ und Bob bestimmt $Y = g^y \bmod p$. Diese Elemente tauschen die beiden miteinander aus und diese können auch von anderen gelernt werden.
Nach dem Austausch berechnet Alice $K = Y^x \bmod p = (g^y)^x \bmod p = g^{xy} \bmod p$ und Bob berechnet $K = X^y \bmod p = (g^x)^y \bmod p = g^{xy} \bmod p$. Damit besitzen beide den gleichen Schlüssel.

b) Alice und Bob möchten einen gemeinsamen Schlüssel erstellen. Dazu einigen Sie sich auf die Werte $g = 6$ und $p = 11$. Alice wählt $x = 4$ und Bob $y = 9$. Berechnen Sie die Werte X und Y und den gemeinsamen Schlüssel K.

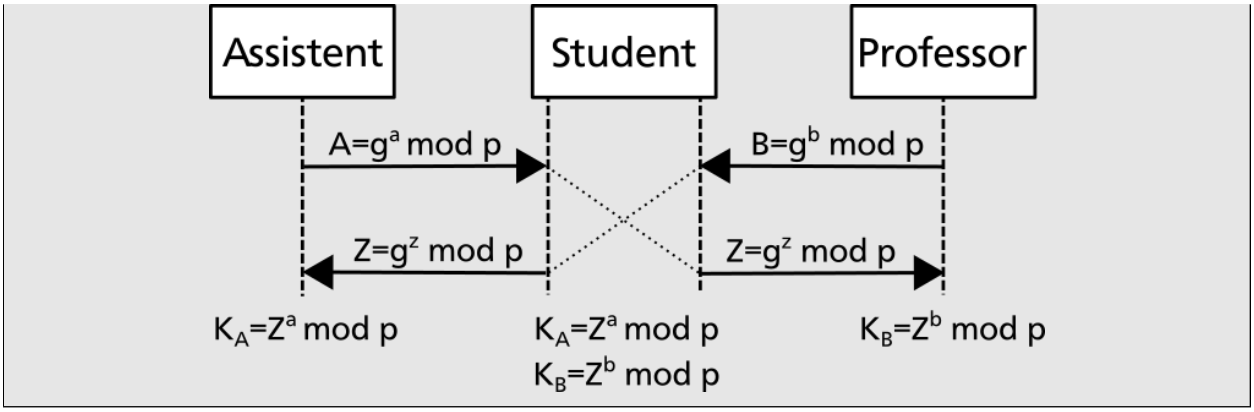
Lösung.
Öffentliche Parameter: $g = 6$, $p = 11$.
Geheime Parameter: Alice wählt $x = 4$ und Bob wählt $y = 9$.
Alice berechnet $X = g^x \bmod p = 6^4 \bmod 11 = 1296 \bmod 11 = 9$ und schickt es an Bob.
Bob berechnet $Y = g^y \bmod p = 6^9 \bmod 13 = 10077696 \bmod 13 = 2$ und schickt es an Alice.
Alice kann danach K berechnen: $K = Y^x \bmod p = 2^4 \bmod 11 = 16 \bmod 11 = 5$.
Bob kann danach ebenfalls K berechnen: $K = X^y \bmod p = 9^9 \bmod 11 = 1387420489 \bmod 11 = 5$.

c) Erklären Sie intuitiv warum ein Angreifer, der die gesamte Kommunikation abhört, nicht an den Schlüssel gelangen kann.

Lösung.
Ein Angreifer, welcher die gesamte Kommunikation abhört, ist im Besitz von X und Y. Außerdem ist ein Angreifer im Besitz der öffentlichen Parameter g und p. Um an den gemeinsamen Schlüssel zu gelangen muss der Angreifer die geheimen Exponenten x und y bestimmen. D.h., dass der Angreifer das Diskrete-Logarithmus-Problem für $g^x \bmod p$ oder $g^y \bmod p$ lösen muss. Den diskreten Logarithmus in modularen Arithmetik (vgl. Foliensatz 2) zu berechnen ist jedoch ein schwieriges Problem, für das bislang keine effizienten Algorithmen bekannt sind, und deshalb kann der Angreifer nicht die Exponenten ermitteln. Die Empfehlung des BSI (Februar 2016) ist, dass p mindestens 2000 Bits groß sein sollte, damit die Berechnung des diskreten Logarithmus nicht in vertretbarer Zeit vollzogen werden kann.

d) Der Assistent von Computersystemsicherheit ist dabei, die Klausur zu entwerfen und möchte einzelne Aufgaben an den Professor schicken. Damit den Studierenden diese Aufgaben nicht in die Hände fallen, möchte er die Kommunikation verschlüsseln. Da der Professor sich aber in einem anderen Gebäude befindet, kann der Schlüsselaustausch nicht persönlich erfolgen, sondern muss digital durchgeführt werden. Aus diesem Grund wendet der Assistent das Diffie-Hellman-Schlüsselaustauschverfahren an. Ein Student hat sich jedoch in das Netzwerk gehackt mit dem Ziel die Nachrichten zwischen dem Professor und Assistenten abzufangen und ggf. zu manipulieren.
Überlegen und formalisieren Sie wie der Student diesen Angriff (nachdem er im Netzwerk ist) ausführt. Wie gelangt der Student in den Besitz des geheimen Schlüssels? Warum kann der Student jetzt Nachrichten manipulieren?

Lösung.
Angenommen der Assistent A und der Professor B möchten miteinander kommunizieren und nutzen das Diffie-Hellman-Schlüsselaustauschverfahren. Dazu einigen sie sich auf ein g und p. Beide wählen jeweils einen geheimen Exponten a und b. Der Assistent berechnet $A = g^a \bmod p$ und der Professor $B = g^b \bmod p$ und tauschen die jeweiligen Werte miteinander aus. Der Student fängt nun die beiden Werte ab. Statt diese weiterzuleiten, wählt er selbst ein z und berechnet $Z = g^z \bmod p$, welches er an den Assistenten und den Professor weiterleitet. Diese glauben nun, dass dieses Z vom jeweiligen Kommunikationspartner kommt und berechnen dementsprechend den Schlüssel $K_A = Z^a$ bzw. $K_B = Z^b$. Da der Student in Kenntnis von A und B ist, kann dieser sowohl $K_A = A^z$, als auch $K_B = B^z$ berechnen und ist somit im Besitz von beiden Schlüsseln. Im weiteren Kommunikationsverlauf fängt der Student alle Nachrichten ab, entschlüsselt sie mit K_A bzw. K_B und verschlüsselt sie mit dem jeweils anderen Schlüssel, bevor er sie zum Kommunikationspartner weiterleitet. So kann die gesamte, eigentlich verschlüsselte Kommunikation abgefangen werden und der Student erfährt, welche Aufgaben in der Klausur drankommen und kann ggf. auch die Aufgaben ändern.



Hausübung. Dieser Bereich ist dazu gedacht das Gelernte weiter zu vertiefen. Dazu werden je nach Themen weitere Übungsaufgaben, ergänzende Beweise oder ähnliche Aufgaben gestellt. Die Aufgaben sind freiwillig, können aber, bei erfolgreicher Bearbeitung, zu Bonuspunkten in der Klausur führen. Die Abgabe dieser Übungen erfolgt über **moodle** und kann in Gruppen mit bis zu vier Studenten (aus Ihrer **eigenen** Übungsgruppe) eingereicht werden. Abgaben werden nur als **.pdf-Dateien** akzeptiert. Denken Sie bitte daran, dass Ihre Lösungen nachvollziehbar und entsprechend ausführlich dargestellt werden sollen.

Für Gruppenabgaben ist folgendes zu beachten: Sie müssen in der Abgabe (.pdf-Datei) deutlich und eindeutig kennzeichnen mit welchen Gruppenpartnern die Aufgaben gelöst wurden.

Der Fachbereich Informatik misst der Einhaltung der Grundregeln der wissenschaftlichen Ethik großen Wert bei. Zu diesen gehört auch die strikte Verfolgung von Plagiarismus. Falls dieser Fall eintritt, behalten wir uns das Recht vor für diese Abgabe den jeweiligen Gruppen keine Punkte gutzuschreiben.

Bitte reichen Sie Ihre Abgabe bis **spätestens Freitag 23.11.2018 um 11:40 Uhr** ein. Verspätete Abgaben können **nicht** berücksichtigt werden.

Hausübung 1 (Diffie-Hellman Schlüsselaustausch (2+2 Punkte)). Alice möchte mit Bob auf sichere Weise kommunizieren. In einer Einführungsveranstaltung zur Kryptographie erfährt Alice von dem Diffie-Hellman Schlüsselaustauschverfahren und möchte dieses dafür verwenden.

a) Sie beobachten folgende Konversation zwischen Alice und Bob.

Alice $\xrightarrow{p = 13\ ,\ g = 6,\ X = 2}$ Bob

Alice $\xleftarrow{Y = 3}$ Bob

Finden Sie den vereinbarten Schlüssel.

Lösung.
Um den gemeinsamen Schlüssel $K = g^{x \cdot y} = (g^x)^y = X^y = (g^y)^x = Y^x$ zu berechnen, müssen wir einen der geheimen Exponenten x oder y kennen. Um z.B. x zu bestimmen, müssen wir den diskreten Logarithmus von $X = g^x \pmod{13}$ lösen. Da die Gruppe $\mathbb{Z}_{13}^*$ nicht sehr groß ist, können wir die Aufgabe leicht lösen, indem wir alle Möglichkeiten für $g = 6$ durchtesten bis wir ein x finden, so dass $g^x \equiv 2 \bmod 13$:

$x = 1:$ $6^1 \equiv 6 \bmod 13.$
$x = 2:$ $6^2 \equiv 36 \equiv 10 \bmod 13.$
$x = 3:$ $6^3 \equiv 36 \cdot 6 \equiv 10 \cdot 6 \equiv 60 \equiv 8 \bmod 13.$
$x = 4:$ $6^4 \equiv 6^3 \cdot 6 \equiv 8 \cdot 6 \equiv 9 \bmod 13.$
$x = 5:$ $6^5 \equiv 9 \cdot 6 \equiv 2 \bmod 13.$

So können wir ablesen, dass, wenn $X = 2 \equiv 6^x \pmod{13}$ ist, $x = 5$ gelten muss. Damit können wir den vereinbarten Schlüssel $K = Y^x = 3^5 \equiv 9 \pmod{13}$ berechnen. Zur Probe kann man auch noch y berechnen und zeigen, dass X^y ebenfalls denselben Schlüssel ergibt.

b) Alice entscheidet sich um ihren Rechenaufwand zu reduzieren eine additive Gruppe für den Diffie-Hellman-Schlüsselaustausch zu verwenden, d.h. statt $X = g^a$ verwendet sie jetzt $X = g \cdot a$.
Begründen Sie, warum das keine gute Idee ist.

Lösung.
Da das Diskrete Logarithmus Problem in additiven Gruppen (gegeben: $A = g \cdot a$, finde: a) mit dem erweiterten Euklidischen Algorithmus, der eine polynomielle Laufzeit besitzt, leicht lösbar ist[1], bietet dass Diffie-Hellman Schlüsselaustauschverfahren in additiven Gruppen keine Sicherheit.
Mathematische Randnotiz warum der erweiterte Euklidische Algorithmus hier immer anwendbar ist: Jeder Generator g von $(\mathbb{Z}_m, +)$ muss bereits die Anforderung ggT(m, g) = 1 erfüllen, denn falls ggT(m, g) = d > 1, so ist die Gruppe $|\langle g \rangle| = m/d < m$, was ein Widerspruch dazu wäre, dass g die Gruppe $\mathbb{Z}_m$ erzeugt. Damit ist der erweiterte Euklidische Algorithmus immer anwendbar.

Hausübung 2 (RSA (2 Punkte)). Letztes Semester haben nur die folgenden 3 Studenten an der Klausur zur Vorlesung "Unsichere Kryptosysteme" teilgenommen:

Name	Matrikelnummer
Alice	174458
Bob	217632
Charlie	224710

Die Menge an möglichen erreichbaren Noten war

$$\{1.0, 1.3, 1.7, 2.0, 2.3, 2.7, 3.0, 3.3, 3.7, 4.0, 5.0\}.$$

Nach der Klausur wurden die Noten mittels des RSA-Verfahrens verschlüsselt und ans Studienbüro weitergeleitet. Der benutzte public key ist $pk = (N, e) = (111791377, 3)$. Für jeden Teilnehmer

[1] Wir haben z.B. in Aufgabe 4b) gesehen, dass man mit dem erweiterten Euklidischen Algorithmus leicht multiplikativ inverse Elemente berechnen kann.

wird die jeweilige Matrikelnummer als $x_1x_2x_3x_4x_5x_6$ interpretiert und mit der erreichten Note $y_1.y_2$ zu dem String $x_1x_2x_3x_4x_5x_6y_1y_2$ codiert. Schließlich wird dieser mit dem obigen public key verschlüsselt und der Ciphertext übermittelt.[2] Folgende Ciphertexts wurden ans Studienbüro geschickt:

$$106894622, \quad 54756549, \quad 49966544$$

In jedem dieser Ciphertexte steht jeweils die Note eines Teilnehmers verschlüsselt und diese Reihenfolge muss nicht mit der obigen Tabelle übereinstimmen.
Obwohl es im Allgemeinen als schwierig gilt die RSA-Verschlüsselung ohne weitere Kentnisse zu invertieren ist es in diesem Fall relativ einfach herauszufinden welche Note Alice in der Klausr erreicht hat. Welche Note hatte sie?

Lösung.
Die Menge der möglichen Klartexte ist sehr klein. Deshalb versuchen wir nicht den gegebenen Ciphertext zu entschlüsseln, sondern wir verschlüsseln alle möglichen Kombinationen aus Alices Matrikelnummer und den möglichen erreichbaren Noten. Danach können wir alle der so ermittelten Ciphertexte mit dem übermittelten Ciphertext vergleichen und können damit bestimmen welcher Klartext dem übermittelten Ciphertext zugrunde liegt. Damit wissen wir dann die Note von Alice.
Wenn man alle Werte einzeln ausrechnet, erhält man folgende Ciphertexte:

Note	Verschlüsselung
1.0	$c = m^e \bmod N = 17445810^3 \bmod 111791377 = 90171184$
1.3	$c = m^e \bmod N = 17445813^3 \bmod 111791377 = 21871285$
1.7	$c = m^e \bmod N = 17445817^3 \bmod 111791377 = 54756549$
2.0	$c = m^e \bmod N = 17445820^3 \bmod 111791377 = 60593177$
2.3	$c = m^e \bmod N = 17445823^3 \bmod 111791377 = 2381692$
2.7	$c = m^e \bmod N = 17445827^3 \bmod 111791377 = 85982807$
3.0	$c = m^e \bmod N = 17445830^3 \bmod 111791377 = 101909109$
3.3	$c = m^e \bmod N = 17445833^3 \bmod 111791377 = 53787838$
3.7	$c = m^e \bmod N = 17445837^3 \bmod 111791377 = 76315827$
4.0	$c = m^e \bmod N = 17445840^3 \bmod 111791377 = 102333603$
5.0	$c = m^e \bmod N = 17445850^3 \bmod 111791377 = 61872659$

Man beobachtet, dass nur der Wert 54756549, der der Verschlüsselung von Alices Matrikelnummer und der Note 1.7 entspricht, im beobachten Ciphertext vorkommt. Also hat Alice die Klausur mit der Note 1.7 bestanden.

Hausübung 3 (RSA (1+1+2 Punkte)). Alice und Bob haben von dem RSA-Verschlüsselungsverfahren gehört und möchten sich das jetzt gerne etwas genauer ansehen.

a) Gegeben ist ein RSA-Modulus $N = p \cdot q$ mit $N = 77$ und $p = 7, q = 11$ bekannt. Weiterhin ist der public key $pk = (N, e) = (77, 6)$ und Alice möchte mit diesem Schlüssel $m_1 = 5$ und $m_2 = 6$ verschlüsseln. Da Alice aber nicht mehr weiß wie das RSA-Verfahren funktioniert benötigt sie Ihre Hilfe.
Berechnen Sie mit Hilfe des RSA-Verfahrens die Verschlüsselungen von m_1 und m_2.

[2] Angenommen Bob hätte eine 5.0 in der Klausur erreicht dann würde der String 21763250 zu 95684781 verschlüsselt werden.

Lösung.

$c_1 = m_1^e \bmod N = 5^6 \bmod 77 = 71$
$c_2 = m_2^e \bmod N = 6^6 \bmod 77 = 71$

b) Vergleichen Sie die beiden Ciphertexte aus a). Was fällt Ihnen aus? Warum entsteht dieses Ergebnis?

Lösung.

Beim Vergleichen der Ciphertexte fällt auf, dass diese gleich sind ($c_1 \equiv c_2$). D.h., dass zwei verschiedene Klartext auf den gleichen Ciphertext abgebildet werden und dadurch ist eine eindeutige Entschlüsselung nicht mehr möglich. Wir berechnen die Eulersche φ-Funktion für $N = 77$ und erhalten $\varphi(77) = 6 \cdot 10 = 60$. Da $e|\varphi(N)$ (d.h. $6|60$) gibt es kein d, welches die Gleichung $e \cdot d \equiv 1 \bmod \varphi(N)$ erfüllt. Somit ist kein gültiges RSA-Kryptosystem gegeben.

c) Bob besitzt den öffentlichen RSA-Schlüssel $(N, e) = (3127, 6)$ mit $N = pq$.
Warum ist dies kein gültiger öffentlicher RSA-Schlüssel? Ändern Sie den ungültigen Anteil von Bobs öffentlichem Schlüssel minimal so ab, dass er gültig wird, und berechnen Sie anschließend den zugehörigen privaten Schlüssel d für ihn.

Lösung.

Für den öffentlichen Schlüssel muss gelten: $ggT(e, \varphi(N)) = 1$
Bestimme Primfaktorzerlegung von N durch Suchen in der Nähe der Quadratwurzel:
$\Rightarrow n = p \cdot q = 53 \cdot 59$
$\Rightarrow \varphi(N) = (p-1) \cdot (q-1) = 52 \cdot 58 = 3016$
Es gilt $ggT(6, 3016) = 2 \neq 1$, somit ist die Bedingung verletzt.

Um den Schlüssel durch eine minimale Änderung zu korrigieren, muss e korrekt gewählt werden. Hierzu kann die Primfaktorzerlegung von $\varphi(N) = 2^3 \cdot 13 \cdot 29$ betrachtet werden. Jede Primzahl kleiner 3016, die kein Primfaktor von 3016 ist, kann als e verwendet werden. Hier: $e = 7$.
Nun müssen wir d berechnen, so dass $e \cdot d \equiv 1 \bmod \varphi(N)$ gilt. Nach der Anwendung des Erweiterten Euklidischen Algorithmus erhalten wir $d = 431$.