Wolfgang K. Seiler
Tel. 2515

A5, C201

12. Dezember 2016

Modulklausur Kryptologie

• • • **Schreiben Sie bitte auf jedes Blatt Ihren Namen!** • • •

Aufgabe 1: (*7 Punkte*)

a) Was versteht man unter dem „Lawineneffekt“, den eine gute Blockchiffre haben sollte?

Lösung: Moderne Blockchiffren arbeiten in Runden. Von einem Lawineneffekt redet man, wenn die Anzahl der Chiffrebits, die von der Änderung eines einzigen Klartext- oder Schlüsselbits (potentiell) betroffen sind, von Runde zu Runde steigt, bis alle Bits des Ausgabeblocks erreicht sind.

b) Welches sehr sichere klassische Kryptoverfahren verzichtet vollständig auf Diffusion?

Lösung: Der *one time pad* verwendet keinerlei Diffusion; hier hängt jeder Chiffrebuchstabe nur von einem Klartext- und einem Schlüsselbuchstaben ab.

c) Worauf beruht die hohe Sicherheit dieses Verfahrens?

Lösung: Da die Schlüsselbuchstaben rein zufällig gewählt werden müssen, ist auch nach Kenntnis eines Chiffrebuchstaben jeder Klartextbuchstabe für den Kryptanalytiker gleich wahrscheinlich.

d) Welche Anforderungen stellt man an ein kryptographisch sicheres Hashverfahren?

Lösung: Es darf nicht mit realistischem Aufwand möglich sein, zu einem vorgegebenen Hashwert einen Text zu finden, der auf diesen Wert führt. Außerdem muß es praktisch unmöglich sein, zwei Texte zu konstruieren, die auf zum selben Hashwert führen.

Aufgabe 2: (*8 Punkte*)

a) Eine ASCII-Datei aus zehn Tausend Byte deutschem Klartext wird mit DES verschlüsselt. Wie oft muß dazu der DES-Algorithmus ausgeführt werden?

Lösung: DES arbeitet mit Blöcken von 64 Bit, also acht Byte. Die Anzahl der notwendigen DES-Verschlüsselungen ist somit gleich $10\,000/8 = 1\,250$.

b) Welches ist die effizienteste Vorgehensweise, um ohne Kenntnis des Schlüssels an den Klartext zu kommen?

Lösung: Da in rund vierzig Jahren noch niemand einen Angriff gefunden hat, der effizienter ist als das Durchsuchen des Schlüsselraums, dürfte wohl das der beste (und heute auch durchführbare) Ansatz sein. Dabei muß man natürlich nicht mit jeden Schlüssel die gesamte Datei dechiffrieren, denn nach nur wenigen Blöcken wird bei jedem falschen Schlüssel auch für einen Computer klar sein, daß die Entschlüsselung nicht auf deutschen Klartext führt.

c) Speziell für diese Klausur wurde das neue Verfahren DES-Vigenère (kurz DES-V) entwickelt. Sein Schlüssel besteht aus einer natürlichen Zahl n, die üblicherweise zwischen fünf und dreißig gewählt wird, sowie n DES-Schlüsseln $s_1, \ldots, s_n$. Der erste Block der Nachricht wird mit DES und Schlüssel s_1 verschlüsselt, der zweite mit s_2, *usw;* für dem

$(n+1)$-ten Block wird wieder s_1 verwendet, und so weiter. Wie würden Sie vorgehen, um dieses Verfahren zu knacken?

Lösung: Zunächst sollte man durch systematisches Probieren alle Schlüssel bestimmen, die aus dem ersten Chiffreblock etwas machen, was als deutscher Klartext interpretiert werden kann. Um zu sehen, ob eventuell $n = 1$ ist, sollte man diese Schlüssel anhand des zweiten Blocks überprüfen: Falls einer zu einer Entschlüsselung führt, die zusammen mit der des ersten Blocks sinnvollen Text gibt, kann man praktisch sicher sein, daß mit $n = 1$ und dem zugehörige Schlüssel chiffriert wurde, und diese Hypothese läßt sich leicht überprüfen. Ansonsten müssen auch für den zweiten Block alle Schlüssel darauf getestet werden, ob sie etwas liefern, was deutscher Klartext sein könnte. Eventuell kann man auch überprüfen, ob es ein Paar gibt, für das der erste und zweite Block zusammen etwas Sinnvolles liefern. Auch beim dritten, vierten, *usw.* Block überprüft man zunächst mit allen Schlüsseln aus der Liste für den ersten Block, um so gegebenenfalls n zu finden; falls dies erfolgslos bleibt, werden alle Schlüssel getestet. Sobald n gefunden ist, sollte es einfach sein, aus den Schlüssellisten für die ersten n Blöcke das Tupel zu identifizieren, das sinnvollen Klartext liefert.

d) Vergleichen Sie die vier Algorithmen DES-V, Double-DES, Triple-DES mit zwei Schlüsseln und Triple DES mit drei Schlüsseln sowohl bezüglich ihrer Sicherheit als auch bezüglich des Verschlüsselungsaufwands! Stellen Sie dazu jeweils eine begründete Rangliste auf!

Lösung: Am unsichersten ist Double-DES, denn für die *meet in the middle attack* muß der Angreifer nur zweimal den Schlüsselraum für DES durchsuchen, hat also einen Aufwand von ungefähr 2^{57}. DES-V braucht nicht nennenswert mehr, denn hier reicht es, n mal den Schlüsselraum zu durchsuchen, d.h. man braucht $n2^{56}$ DES-Anwendungen. Da man keinen Ergebnisvergleich durchführen muß, geht das insgesamt je nach Hardware vielleicht sogar schneller wie bei Double-DES. Bei Triple-DES mit zwei Schlüsseln müssen alle Schlüssel*paare* getestet werden; der Aufwand ist also etwa $2^{56} \cdot 2^{56} = 2^{112}$. Bei Triple-DES mit drei Schlüsseln schließlich liegt der Aufwand bei 2^{168}, falls man alle Tripel testet; durch eine *meet in the middle attack* kann man das reduzieren auf $2^{112} + 2^{56}$ Verschlüsselungen und Ergebnisvergleich.

Was den Verschlüsselungsaufwand betrifft, so ist DES-V am schnellsten, denn hier muß für jeden Block nur eine DES-Verschlüsselung durchgeführt werden. Bei Double-DES sind es zwei, bei Triple-DES unabhängig von der Anzahl der Schlüssel drei.

Aufgabe 3: *(7 Punkte)*

Die zusammengesetzte Zahl p habe die Eigenschaft, daß $a^p \equiv a \bmod p$ für alle $a \in \mathbb{Z}$.

a) Zeigen Sie: p ist eine CARMICHAEL-Zahl, d.h. für alle zu p teilerfremden ganzen Zahlen a ist $a^{p-1} \equiv 1 \bmod p$.

Lösung: Ist $\mathrm{ggT}(a, p) = 1$, so können wir mit dem erweiterten EUKLIDischen Algorithmus ein $a' \in \mathbb{Z}$ finden, so daß $a' \cdot a \equiv 1 \bmod p$. Damit ist $a^{p-1} \equiv a' \cdot a^p \equiv a' \cdot a \equiv 1 \bmod p$.

b) Man kann zeigen, daß jede CARMICHAEL-Zahl p ein Produkt von mindestens drei verschiedenen Primzahlen p_i ist, wobei $p_i - 1$ für jedes i ein Teiler von $p - 1$ ist. Folgern Sie daraus, daß für jede CARMICHAEL-Zahl p gilt: $a^p \equiv a \bmod p$ für alle $a \in \mathbb{Z}$.

Lösung: Sei $p = p_1 \cdots p_r$. Nach dem kleinen Satz von FERMAT ist $a^{p_i - 1} \equiv 1 \bmod p_i$; für jedes nicht durch p_i teilbare a. Da $p - 1$ ein Vielfaches von $p_i - 1$ ist, gilt dann auch $a^{p-1} \equiv 1 \bmod p_i$, also auch $a^p \equiv a \bmod p_i$. Falls a ein Vielfaches von p_i ist, gilt letztere Kongruenz auch, da dann beide Seiten durch p_i teilbar, also kongruent Null modulo p_i

sind. Somit ist $a^p \equiv a \bmod p_i$ für alle $a \in \mathbb{Z}$. Da dies für jedes p_i gilt und die p_i allesamt verschieden sind, folgt aus dem chinesischen Restesatz (triviale Richtung), daß $a^p \equiv a \bmod p$ für alle $a \in \mathbb{Z}$.

c) Für ein RSA-System mit Modul $N = pq$ werden zwei Zahlen p, q verwendet, für die $a^p \equiv a \bmod p$ und $a^q \equiv a \bmod q$ für alle $a \in \mathbb{Z}$. Zeigen Sie: Sind p und q teilerfremd, und ist e teilerfremd zu $p-1$ und $q-1$, läßt sich stets ein $d \in \mathbb{N}$ finden mit $a^{ed} \equiv a \bmod N$ für alle $a \in \mathbb{Z}$.

Lösung: Da e teilerfremd zu $p-1$ und zu $q-1$ ist, können wir uns über den erweiterten EUKLIDischen Algorithmus natürliche Zahlen d, k verschaffen, für die

$$de - k \cdot kgV(p-1, q-1) = 1 .$$

Die Zahlen p und q sind nach *a)* entweder Primzahlen oder CARMICHAEL-Zahlen. Wir können sie also schreiben als Produkte $p = p_1 \cdots p_r$ und $q = q_1 \cdots q_s$ von Primzahlen p_i und q_j, wobei im Falle von Primzahlen p oder q einfach $p = p_1$ oder $q = q_1$ ist. Da alle $p_i - 1$ Teiler von $p-1$ sind und alle $q_j - 1$ Teiler von $q-1$, ist $m = k\,kgV(p-1, q-1)$ ein Vielfaches aller $p_i - 1$ und aller $q_j - 1$. Nach dem kleinen Satz von FERMAT ist daher

$$a^{de} = a^{1+m} = a \cdot a^m \equiv a \cdot 1 = a \bmod p_i, q_j$$

für alle i, j. Da p und q teilerfremd sind, kann kein p_i gleich einem q_j sein; daher ist auch $a^{de} \equiv a$ modulo dem Produkt $N = pq$ aller p_i und aller q_j.

d) Welche Auswirkungen hat es auf die Sicherheit des Systems, wenn p und q keine Primzahlen sind?

Lösung: Bei einem guten RSA-Modul ist $N = pq$ mit zwei Primzahlen in der Größenordnung von $\sqrt{N}$. Falls eine von diesen durch eine CARMICHAEL-Zahl ersetzt wird, gibt es einen Primteiler von N der deutlich kleiner ist und daher zumindest im Prinzip einfacher zu finden. Die Faktorisierung des Rests ist dann auch zumindest im Prinzip einfacher als die eine Produkts zweier Primzahlen mit gleicher Größenordnung. Somit wird das Verfahren dadurch unsicherer.

Aufgabe 4: (*5 Punkte*)

a) Nachrechnen zeigt, daß $20192^2 \equiv 1 \bmod 42037$. Warum folgt daraus ohne jede weitere Rechnung, daß $N = 42037$ keine Primzahl sein kann?

Lösung: Modulo einer Primzahl p hat die Gleichung $X^2 \equiv 1 \bmod p$ nur die Lösungen $x_1 \equiv 1 \bmod p$ und $x_2 \equiv -1 \bmod p$, da $X^2 - 1$ über dem Körper $\mathbb{F}_p$ als quadratisches Polynom nicht mehr als zwei Nullstellen haben kann. $20192 \bmod N$ ist offensichtlich weder $+1$ noch -1, also kann N keine Primzahl sein.

b) Zerlegen Sie N in seine Primfaktoren!

Lösung: Für jeden Primfaktor p von N ist $20192^2 \equiv 1 \bmod p$, also ist $20192 \equiv \pm 1 \bmod p$, d.h. 20192 ∓ 1 ist durch p teilbar. Dabei kann nicht für alle p das gleiche Zeichen gelten, denn sonst wäre $20192 \equiv \pm 1 \bmod N$. Daher sind die Zahlen $ggT(20192 \pm 1, N)$ echte Faktoren von N.
Wir wenden den EUKLIDischen Algorithmus an auf N und $20192 - 1$:

$$\begin{aligned} 42037 : 20191 &= 2 \quad \text{Rest } 1655 \\ 20191 : 1655 &= 12 \quad \text{Rest } 331 \\ 1655 : 31 &= 5 \quad \text{Rest } 0 \end{aligned}$$

Damit ist 331 ein Teiler von N, und $42037/331 = 127$ ist ein anderer. Beide sind Primzahlen, denn sie sind offensichtlich nicht durch $2, 3$ oder 5 teilbar, $127 \equiv 1 \bmod 7$ und $331 \equiv 2 \bmod 7$, $127 \equiv 6 \bmod 11$ und $331 \equiv 1 \bmod 11$. Da $13^2 > 127$ ist klar, daß 127 prim sein muß. Für 331 müssen wir noch nachrechnen, daß $331 \equiv 6 \bmod 13$ und $331 = 340 - 9 \equiv -9 \bmod 17$. Da $19^2 > 331$, ist auch 331 prim.

Aufgabe 5: (*10 Punkte*)

a) Bestimmen Sie für das RSA-System mit Modul $N = 60701 = 601 \cdot 101$ die kleinste Zahl $e > 20$, zu der es ein $d \in \mathbb{N}$ gibt, so daß $(a^e)^d \equiv a \bmod N$ für alle $a \in \mathbb{Z}$!

Lösung: $p - 1 = 600$ und $q - 1 = 100$ haben das kgV $600 = 2^3 \cdot 3 \cdot 5^2$; somit darf e durch keine der Primzahlen $2, 3, 5$ teilbar sein. Die kleinste Zahl $e > 20$ mit dieser Eigenschaft ist $e = 23$.

b) Bestimmen Sie für diesen Wert des öffentlichen Exponenten e einen privaten Exponenten d!

Lösung: Dazu können wir den erweiterten EUKLIDischen Algorithmus anwenden auf e und das kgV 600 von $(p-1)$ und $(q-1)$:

$$\begin{aligned} 600 : 23 = 26 \quad \text{Rest } 2 &\Longrightarrow 2 = 600 - 26 \cdot 23 \\ 23 : 2 = 11 \quad \text{Rest } 1 &\Longrightarrow 1 = 23 - 11 \cdot (600 - 26 \cdot 23) = 286 \cdot 23 - 11 \cdot 600 \end{aligned}$$

Somit können wir $d = 286$ wählen.

c) Verschlüsseln Sie die Nachricht $m = 2$ mit diesem System!

Lösung: $2^{10} = 1024$, also ist $2^{20} = 1024^2 = 1048576 \equiv 16659 \bmod N$ und

$$2^{23} \equiv 8 \cdot 16659 = 133272 \equiv 11870 \bmod N\,.$$

Die Nachricht wird also als 11870 verschlüsselt.

d) Wenn Sie d als öffentlichen Exponenten veröffentlichen und den in *a)* bestimmten Wert e geheimhalten, können Sie mit deutlich geringerem Aufwand Dokumente unterschreiben. Warum sollten Sie das lieber nicht tun?

Lösung: Ein geheimer Exponent 23 wäre deutlich kleiner als $\sqrt[4]{N}$, so daß er über die Kettenbruchentwicklung von N/d bestimmt werden könnte; im Falle eines so kleinen Wertes könnte man ihn sogar durch Probieren ermitteln.

e) Warum sollte man RSA bei der Verschlüsselung nie in Reinform benutzen, sondern jeden Block mit zufälligen oder durch einen geeigneten Algorithmus bestimmten Bits beginnen lassen?

Lösung: Oft werden mit RSA Nachrichten übermittelt, die deutlich kürzer sind als die Blocklänge, zum Beispiel Schlüssel für ein asymmetrisches Kryptoverfahren. Falls man solche Nachrichten mit führenden Nullen auffüllt, also direkt in eine Zahl umwandelt, gibt es Angriffstechniken, die mit einer unter Sicherheitsaspekten viel zu hohen Wahrscheinlichkeit zur Dechiffrierung des Blocks führen können.

Aufgabe 6: (*13 Punkte*)

a) Zeigen Sie, daß die Zwei in $(\mathbb{Z}/101)^\times$ die Ordnung 100 hat! *(Hinweis: Sie können einiges an Rechenarbeit sparen, wenn Sie gelegentlich auch mit negativen Zahlen arbeiten!)*

Lösung: Wenn die Zwei Ordnung 100 hat, ist natürlich $2^{100} \equiv 1 \bmod 101$. Aus dieser Kongruenz für sich alleine folgt allerdings nur, daß die Ordnung der Zwei ein Teiler von 100 ist. Falls sie ein echter Teiler ist, muß sie Teiler von $100/2 = 50$ oder $100/5 = 20$ sein. Wenn wir zeigen können, daß

$$2^{100} \equiv 1 \bmod 101, \quad 2^{50} \not\equiv 1 \bmod 1 \quad \text{und} \quad 2^{20} \not\equiv 1 \bmod 101$$

gilt, muß die Zwei Ordnung 100 haben. Berechnen wir also schrittweise diese Zahlen: $2^{10} = 1024 \equiv 14 \bmod 101$, also ist $2^{20} \equiv 14^2 = 196 \equiv -6 \neq 1 \bmod 101$.

$$2^{50} = 2^{20} \cdot 2^{20} \cdot 2^{10} \equiv (-6) \cdot (-6) \cdot 14 = 504 \equiv -1 \bmod 101\,;$$

somit ist 2^{50} von eins verschieden, hat aber das Quadrat 2^{100} gleich eins modulo 101.

b) Folgern Sie daraus, daß 101 prim ist!

Lösung: Da es ein Element der Ordnung 100 gibt, ist die Ordnung der Gruppe $(\mathbb{Z}/101)^\times$ mindestens hundert. Da $\mathbb{Z}/101$ aus 101 Elementen besteht, von denen eines, die Null, garantiert nicht invertierbar ist, heißt dies, daß jedes von Null verschiedene Element invertierbar ist, d.h. jede nicht durch 101 teilbare ganze Zahl ist teilerfremd zu 101. Das ist nur möglich, wenn 101 prim ist.

c) Was ist $37^{90807060504030201} \bmod 101$?

Lösung: Da 101 prim ist, gilt nach dem kleinen Satz von Fermat $a^{100} \equiv 1 \bmod 101$ für alle nicht durch 101 teilbaren ganze Zahlen; insbesondere ist also $37^{100} \equiv 1 \bmod 101$ und $37^x \bmod 101$ hängt nur ab von $x \bmod 100$. Für den Exponenten x aus der Aufgabenstellung ist $x \bmod 100 = 1$, also ist $37^x \equiv 37 \bmod 101$.

d) Zeigen Sie: Ist x ein diskreter Logarithmus von a zur Basis zwei modulo 101, so ist $x + 50$ ein diskreter Logarithmus von $-a$.

Lösung: Nach obiger Rechnung ist $2^{50} \equiv -1 \bmod 101$; falls $2^x \equiv a \bmod 101$, folgt also $2^{x+50} = 2^x \cdot 2^{50} \equiv a \cdot (-1) = -a \bmod 101$.

e) Bestimmen Sie den diskreten Logarithmus modulo 101 von 6 zur Basis 2! *(Hinweis: Wenn Sie Ihre Rechenergebnisse aus a) verwenden, finden Sie den sehr schnell.)*

Lösung: Nach der Rechnung zum Nachweis von *a)* ist $2^{20} \equiv -6 \bmod 101$. Aus *d)* folgt daher, daß $2^{70} \equiv 6 \bmod 101$, d.h. der gesuchte diskrete Logarithmus ist siebzig.

f) Wie funktionieren elektronische Unterschriften nach DSA?

Lösung: Zunächst wird eine Primzahl q gewählt mit, nach heutigen Standards, mindestens 256 Bit; danach eine Primzahl p mit $p \equiv 1 \bmod q$, die derzeit mindestens 2048 Bit haben sollte. Der Unterschreibende wählt einmalig ein Element $g \in (\mathbb{Z}/p)^\times$ der Ordnung q sowie einen geheimen Schlüssel x; er veröffentlicht p, q, g und $u = (g^x \bmod p) \bmod q$.
Um mit diesem Verfahren einen Hashwert $m < q$ zu unterschreiben, wählt er für jede Nachricht neu eine Zufallszahl $k < q$ und berechnet $r = (g^k \bmod p) \bmod q$. Da q eine Primzahl ist, hat k ein multiplikatives Inverses $k^{-1} \bmod q$. Damit läßt sich ein $s < q$ bestimmen, so daß $sk \equiv m + xr \bmod q$ ist. Die Unterschrift ist das Paar (r, s).
Mit $t = s^{-1} \bmod q$ ist $k \equiv tsk \equiv tm + xtr \bmod q$; da g in $\mathbb{F}_p^\times$ die Ordnung q hat, ist $g^k \equiv g^{tm} g^{xtr} \equiv g^{tm} u^{tr} \bmod p$. Modulo q ist daher bei einer korrekten Unterschrift

$$r = (g^k \bmod q) = \big((g^{tm} u^{tr}) \bmod p\big) \bmod q\,.$$

Die Unterschrift wird anerkannt, wenn diese Kongruenz erfüllt ist.

Aufgabe 7: *(6 Punkte)*

a) Lösen Sie im Körper $\mathbb{F}_{17}$ die Gleichung $6x = 10$!

Lösung: Dazu muß zunächst das multiplikative Inverse von 17 bestimmt werden, d.h. wir müssen den ggT 1 von 6 und 17 als Linearkombination dieser beiden Zahlen darstellen.

$$17 : 6 = 3 \quad \text{Rest } -1 \Longrightarrow -1 = 17 - 3 \cdot 6 \Longrightarrow 1 = 3 \cdot 6 - 17\,.$$

Somit ist $6^{-1} = 3$ in $\mathbb{F}_{17}$. Multiplizieren wir die Gleichung damit, erhalten wir

$$x = 3 \cdot 10 = 30 \equiv 13 \bmod 17\,.$$

(Wir hätten auch zunächst kürzen können zu $3x = 5$; dann hätten wir mit dem Inversen 6 von 3 multipliziert und wären auf das gleiche Ergebnis gekommen.)

b) Berechnen Sie dort mit der *baby step – giant step* Methode eine Lösung der Gleichung $6^x = 10$!

Lösung: Wir brauchen zunächst eine natürliche Zahl m knapp über $\sqrt{17}$; hier bietet sich $m = 5$ an. In den *baby steps* berechnen wir die Potenzen von 6 mit Exponenten kleiner m: In $\mathbb{F}_{17}$ ist

$$6^1 = 6, \qquad 6^2 = 2, \qquad 6^3 = 12, \qquad 6^4 = 2^2 = 4\,.$$

Für die *giant steps* berechnen wir $10 \cdot 6^{-mj}$ so lange, bis wir einen Wert aus obiger Liste erreicht haben. Dazu brauchen wir zunächst 6^{-m}. Wie wir aus *a)* wissen, ist $6^{-1} = 3$, und $3^m = 3^5 = 3^3 \cdot 3^2 = 10 \cdot 9 = 5$. Wir beginnen also mit $10 \cdot 5 = -1 = 16$, was nicht in obiger Liste steht. Multiplikation mit fünf liefert

$$10 \cdot 6^{-2m} = -1 \cdot 5 = -5 = 12 = 6^3\,.$$

Somit ist $10 \cdot 6^{-10} = 6^3$, also $10 = 6^{13}$. Die Lösung ist $x = 13$. (Es ist „Zufall“, daß dies mit der Lösung von *a)* übereinstimmt; allgemeine Schlüsse lassen daraus nicht ziehen.)

Aufgabe 8: *(4 Punkte)*

a) Beschreiben Sie das Verfahren zum Schlüsselaustausch nach Diffie und Hellman!

Lösung: Die beiden Beteiligten einigen sich auf eine große Primzahl p und ein Element $a \in \mathbb{F}_p^\times$ mit möglichst großer Ordnung, idealerweise eine primitive Wurzel. p sollte nach heutigen Sicherheitsstandards etwa 2048 Bit haben. Dann wählt jeder eine geheimzuhaltende Zahl x *bzw.* y aus dem Intervall von eins bis $p - 2$; er schickt a^x *bzw.* a^y an den jeweils anderen. Mit ihrer geheimen Information x *bzw.* y können dann beide das Element $a^{xy} = (a^x)^y = (a^y)^x$ berechnen, das als Grundlage eines zu vereinbarenden Schlüssels verwendet werden kann.

b) Gegen welche Art von gegnerischem Angriff muß man sich hier vor allem schützen?

Lösung: Am gefährlichsten ist der Angriff durch einen *man in the middle,* das heißt, ein Gegner überwacht und unterbricht die Kommunikation zwischen den beiden Beteiligten und gibt sich gegenüber jedem von ihnen als der jeweils andere aus. Auf diese Weise kann er mit jedem der beiden einen Schlüssel vereinbaren. Beim anschließenden Nachrichtenaustausch entschlüsselt er die aufgefangenen Nachrichten mit dem zugehörigen Schlüssel und sendet sie (eventuell verändert) nach Verschlüsselung mit dem anderen Schlüssel weiter. Die Beteiligten müssen daher unbedingt sicherstellen, daß am anderen Ende der Übertragungsstrecke wirklich die angenommene Person (oder deren Computer) sitzt.