Krypto-logisch

Prof. Dr. Dörte Haftendorn, Universität Lüneburg, Dillingen GDM September 2005

Kurzfassung

Ohne PIN-Nummern, sicheren Datentransfer, digitale Signatur u.a. ist unsere Welt nicht mehr denkbar. Die moderne Kryptografie beruht auf Berechnungen modulo großer Primzahlen oder Primzahlprodukten.
Sie hat ihre Wurzeln damit in Algebra und Zahlentheorie, ist aber schon mit überschaubaren Primzahlen ohne Computer nicht zu bewältigen. Zentrale algorithmische Anforderungen liegen beim erweiterten Euklidischen Algorithmus und beim Potenzieren im Modul. Informatische Aspekte sind also die Entwicklung von entsprechenden Funktionen. Die großen CAS können das, für den TI-voyage werden Lösungen vorgestellt. Auch die Abarbeitung eines kryptografischen Protokolls ist ein Algorithmus im klassischen Sinn.

Der Vortrag beruht auf Erfahrungen im Informatikunterricht des Gymnasiums und in Vorlesungen für Lehramtsstudierende. Für letztere dient die Kryptographie als Ziel und Sinngebung für die Themen "Algebra und Zahlentheorie". Es ist faszinierend wie hier ein gesellschaftlich außerordentlich wichtiges Thema in schulisch überschaubarem mathematischen Handeln transparent wird.
http://www.mathematik-verstehen.de → Kryptographie

##

Mathematische Grundlagen des RSA-Verfahrens.

In der Kryptografie werden Primzahlen in der Größenordnung von 150 dezimalen Stellen verwendet. Die Sicherheit der Verfahren beruht darauf, dass das Produkt aus zwei solchen Primzahlen nicht effektiv wieder zerlegt werden kann. Im Folgenden wird das Vorgehen mit kleinen Primzahlen verdeutlicht. Sei $p=11,\ q=13,\ n=pq=143$. Aus den Restklassen modulo n werden die zu n Teilerfremden zu der Menge $\mathbb{Z}_n^*$ zusammengefasst, mit der Multiplikation modulo n bilden diese die „prime Restklassengruppe“. Sie enthält $ord(\mathbb{Z}_n^*)=\varphi(n)=(p-1)(q-1)$ Elemente. Hier fallen aus den Zahlen bis 143 die 11-Vielfachen und die 13-Vielfachen weg, es bleiben $\varphi(143)=10\cdot 12=120$ Teilerfremde. In allen endlichen Gruppen G gilt $a\in G\Rightarrow a^{ord(G)}=1$, also hier $a\in\mathbb{Z}_n^*\Rightarrow a^{\varphi(n)}=1$ (Eulerscher Satz, Begr. s.u.). Daher kann man in den Exponenten modulo φ rechnen. Das reduziert die Exponenten in der Größe, denn $a^{247}\underset{143}{\equiv}a^{7}$. Viel wichtiger für die Kryptografie ist aber, dass die Exponenten, die teilerfremd zu φ sind, Inverse modulo φ haben. In diesem Beispiel gilt für alle $a\in\mathbb{Z}_n^*$ nämlich $a^{7\cdot 103}\underset{143}{\equiv}a^{721}\underset{143}{\equiv}\left(a^{120}\right)^6 a^1=1^6\cdot a=a$. Nun wählt man also beim RSA-Verfahren einen „öffentlichen Schlüssel“ $e\in\mathbb{Z}_\varphi^*$. Weil $\left(\mathbb{Z}_\varphi^*,\bullet\right)$ eine Gruppe ist, gibt es das Inverse $d\in\mathbb{Z}_\varphi^*$ mit $e\,d=1\ in\ \mathbb{Z}_\varphi^*$. Also gilt für eine Nachricht $m\in\mathbb{Z}_n^*$ immer $m^{e\,d}=m\ in\ \mathbb{Z}_n^*$

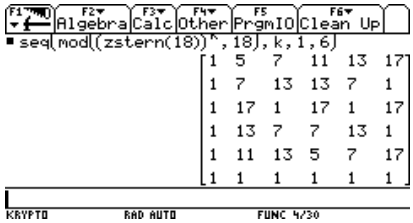
Man beschafft d aus e und φ mit dem erweiterten euklidischen Algorithmus , der die „Vielfachsummendarstellung“ erzeugt $1=d\ e+t\ \varphi\underset{\varphi}{\equiv}d\ e$. Man veröffentlicht e und n und hält d geheim.

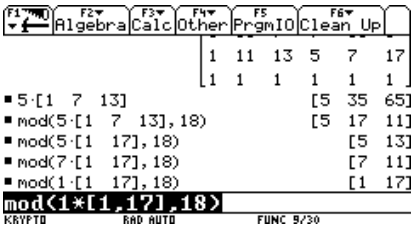
Im RSA-Protokoll ist $c = m^e$, $m \in \mathbb{Z}_n^*$ die verschlüsselte Nachricht (message) , die der Empfänger durch Potenzierung mit dem geheimen d entschlüsseln kann.

$c^d = \left(m^e\right)^d = m^{ed} = m^1 = m$. Bei der digitalen Signatur wird von dem Signierenden mit d potenziert, von dem Verifizienden mit dem öffenlichen Schlüssel e. Alle Rechungen finden modulo n statt.

Didaktische Aspekte

Je nachdem, in welchen Zusammenhang man eine Lehreinheit zur Kryptografie stellen möchte, ergeben sich natürlich jeweils andere Schwerpunkte.
Ganz sicher aber müssen Primzahlen, Teilbarkeit und ggT thematisiert werden. Bei letzteren ist die Beschaffung aus dem Euklidischen Algorithmus unerlässlich, da man mit ihm durch Rückwärtsarbeiten an die Vielfachsummendarstellung herankommt. Diesen Algorithmus sollen die Lernenden „von Hand“ können, aber für eine sinnvollen Arbeit mit kryptografischen Protokollen muss unbedingt ein CAS eingesetzt werden (s.u.).
Als zweites Standbein der Kryptografie ist das Rechnen in Restklassenringen zu erarbeiten. Erfahrungsgemäß wird das schnell verstanden. Allgemeinere Stukturüberlegungen sind nicht unbedingt nötig, gehören aber in der Lehrerausbildung sicher dazu.
Der Begriff „Ordnung eines Gruppenelementes a“ als kleinste Zahl k mit $a^k = 1$ kann ebenso wie die Tatsache, dass die Elementordnung die Gruppenordnung teilt, aus der Betrachtung von Potenzen-Tafeln der Primen Restklassengruppe $\mathbb{Z}_n^*$ gewonnen werden. Bildet man die Nebenklassen $g \cdot \langle a \rangle$ der von einem Element a erzeugten Untergruppen und überlegt, dass sie alle gleiche Elementzahl haben, dann ist der Eulersche Satz (s.o.) bewiesen. Rückgriffe auf gruppentheoretische Ergebnisse , die den Lernenden nicht einleuchten, sind – entgegen häufiger Formulierung in Büchern -- nicht nötig.





Mit diesem „Rüstzeug“ können mindestens fünf wichtige kryptografische Verfahren verstanden werden, ebenso auch digitale Signatur, Zertifizierungen und elektronisches Geld. Dies ist eine hinreichend breite Palette, die auch Klausuren u.ä. erlaubt.

Algorithmische Aspekte im Hinblick auf die Lehre

Der erweiterte Euklidische Algorithmus ist sowieso lehrreich, muss aber hier als CAS-Befehl vorliegen. Für der TI voyage hat die Autorin ein Progamm entwickelt (siehe Internet), ein Informatikkurs könnte dies als eine Aufgabe bekommen.
Durchaus algorithmisch interessant sind ja auch schon die Erzeugung passender Mengen, Folgen und Tafeln wie oben bei der Potenzentafel gezeigt.
Auch zu Primzahltest sollten zumindest Überlegungen angestellt werden.

Die Sicherheit der modernen Kryptografie beruht ja auf der Unmöglichkeit, Zahlen von weit über 200 Stellen effektiv in ihre Faktoren zu zerlegen. Da muss man auch anders als man es bei kleinen Zahlen macht entscheiden können, ob eine 150 Stellen lange Zahl Primzahl ist oder nicht. Schulisch leicht erreichbar ist der „Kleine Fermatsche Satz“ als Spezialfall des Eulerschen Satzes (s.o.) $p \; sei \; prim, a < p \Rightarrow \varphi(p) = (p-1) \Rightarrow a^{p-1} \underset{p}{\equiv} 1$. Diese letzte (notwendige aber nicht hinreichende) Bedingung ist algorithmisch und auch von den Lernenden leicht zu prüfen. Ist sie verletzt, kann p keine Primzahl sein.

Als ganz wichtig erweist sich ein geschickter Umgang mit der Potenzierung im Modul, denn auch bei recht kleinen Zahlen sprengen die Potenzen schnell den Bereich für eine exakte

Zahldarstellung. In obigem Beispiel könnte $127^{103} \bmod 143$ vorkommen und es ist klar, dass man nicht erst die 216 dezimalen Stellen der Potenz berechnen sollte, bevor man den Rest modulo 143 bestimmt. Die CAS halten den Befehl „Powermod“ bereit, für den TI voyage muss er programmiert werden. Die Idee ist, iterativ zu quadrieren, stets sofort modular herunterzubrechen und nur gewisse Zwischenergebnisse als Faktoren in die sich so bildende Potenz aufzunehmen. Und zwar denkt man sich den Exponenten als Dualzahl geschrieben, die man rechts beginnend durchgeht, und man fügt genau dann ein, wenn man eine 1 liest.
Des Weiteren muss die Überführung einer Wortnachricht in eine Zahl (und zurück) thematisiert werden. Es eignet sich eine geschickte Ausnutzung des ASCII-Codes.

Die kryptographischen Verfahren selbst sind Algorithmen im klassischen Sinn. Sie haben meist eine Schlüsselerzeugungsphase und eine Anwendungsphase. Manchmal, wie zum Beispiel beim Diffie-Hellmann-Verfahren, wird ein Schlüssel erzeugt, der dann für ein ganz anderes Verfahren verwendet wird. Jedenfalls ist die schrittweise Duchführung „von Hand“ unterrichtlich unentbehrlich. „Von Hand“ heißt hier mit einem Werkzeug, dass „ggt-erweitert“ und „powermod“ ausführen kann.
Als nächste Stufe sind kommentierte CAS-Dateien oder Tutorskripten beim TI voyage hilfreich. (Siehe Internet) Ihre Erstellung ist auch als Aufgabe höchst sinnvoll.
Zu den klausurfähigen Kompetenzen sollte die Interpretation und die Erstellung solcher Texte gehören. Auch eine graphische Verdeutlichung, wer was rechnet, was wem bekannt ist, wer was wem schickt u.s.w. ist alles andere als trivial.
Ein Informatikkurs könnte auch ein Programm erstellen, das das Verfahren vollständig „durchzieht“. Für alle anderen Lernenden halte ich das Arbeiten mit fertigen „Tools“ zumindest für den Anfang nicht für so sinnvoll, da dabei das Vorgehen zu stark verborgen wird. (Z.B. www.cryptool.de der Universitäten Siegen und Darmstadt). Hat man aber für das oben Dargestellte zeitlich oder curricular gar keinen Platz, so ist die erläuterte Anwendung so eines Tools noch um ein Vielfaches besser als die Vermeidung dieses Themas.

Die klassische Kryptografie (Alphabet-Verschiebungen u.ä.) ist allenfalls für sehr junge Lernende oder als Einstieg sinnvoll. Es handelt sich ausschließlich um Historie. Man mache sich klar, dass die mit moderner Kryptografie verschlüsselten Texte keinerlei „Muster“ aufweisen, die man irgendwie doch herauskriegen könnte. Angreifen kann man ausschließlich mit mathematischen Vorgehensweisen in Modulringen.

Gesellschaftliche Aspekte

„Kryptografie umgibt uns“ könnte man pointiert sagen. Jedenfalls ist kein anderes mathematisches Teilgebiet in unser Welt so allgegenwärtig. Die gute alte Geldbörse mutiert schon zur einer Kartenbibliothek, jede solche Karte kommuniziert in der Sprache der Kryptografie mit dem Automaten, in den sie gesteckt wird, dieser sendet und empfängt verschlüsselte Daten von seinem Auftraggeber. Beim Handy-Telefonieren, beim Interneteinkauf, beim Online-Banking, bei der Software - überall geht es um Autentifizierung, sicheren Datentransport, Zertifizierung – um kryptografische Anwendungen.

Es wird Zeit, dass sich jede mathematische Ausbildung diesen Fragen stellt, insbesondere dürfte es heute keine Mathematiklehrerausbildung mehr ohne Kryptografie geben. Mindestens die Mathematik-Lehrenden müssen die Kompetenz erwerben, das Thema altersgemäß und verständlich zu unterrichten, die Lernenden handelnd einzubeziehen und ihnen das Gefühl zu vermitteln, für’s Leben etwas gelernt zu haben.