Seminararbeit

Cryptanalytic Attacks on RSA

Quantum Computing Attack

Eingereicht am:

5. Juni 2016

Eingereicht von:
Rimbert Fischer
Matrikelnummer: inf100606
E-Mail: inf100606 (at) fh-wedel.de

Referent:
Prof. Dr. Gerd Beuster
Fachhochschule Wedel
Feldstraße 143
22880 Wedel
E-Mail: gb (at) fh-wedel.de

Inhaltsverzeichnis

1 Einleitung

Die Sicherheit des Verschlüsselungsverfahrens RSA beruht bisher immer darauf, dass alle Methoden und Algorithmen viel zu lange dauern würden, mit RSA verschlüsselte Nachrichten effizient zu entschlüsseln. Aber was wäre, wenn es einen Computer gäbe, der entsprechende Nachrichten mit entsprechenden Algorithmen in kurzer Zeit entschlüsseln könnte? Quantencomputer sind zwar noch nicht ausreichend dazu entwickelt, haben aber das Potential, dies zu schaffen. Aufgrund der Tatsache, dass RSA ein sehr verbreitetes Verschlüsselungsverfahren ist, hätte die Entschlüsselung von RSA in angemessener Zeit verheerende Folgen.

2 Mathematische Grundbegriffe

2.1 Elementordnung

In der Gruppentheorie ist die Ordnung r eines Elements x in einer Gruppe G die kleinste natürliche Zahl $r > 0$, für die gilt:

$$x^r \equiv 1 \ (mod\ N)$$

Die Elementordnung wird auch als Periode bezeichnet und dargestellt mit der Funktion:

$$order(r, N)$$

2.2 Bra-Ket-Notation

Zur Darstellung der Zustandsvektoren in der Quantenmechanik wird die Bra-Ket-Notation verwendet:

$$|\mathrm{v}\rangle = \begin{pmatrix} v_1 \\ \vdots \\ v_n \end{pmatrix}$$

3 RSA Allgemein

RSA ist ein asymmetrisches Verfahren zum Verschlüsseln und Signieren von Nachrichten. Im Folgenden wird RSA als Verschlüsselungsverfahren betrachtet. RSA besteht aus einem öffentlichen Schlüssel zum Verschlüsseln und einem privaten Schlüssel zum Entschlüsseln der Nachricht. Die Erstellung der Schlüssel, sowie das Ver- und Entschlüsseln wird hier kurz erklärt.

Es werden zwei große Primzahlen p und q benötigt. Aus diesen wird das RSA-Modul N berechnet:

$$N = p \cdot q$$

Anwendung der Eulerschen φ-Funktion auf N:

$$\varphi(N) = (p-1)(q-1)$$

Wählen eines e, welches teilerfremd ist zu $\varphi(N)$ und für das gilt:

$$1 < e < \varphi(N)$$

Berechnung von d als multiplikatives Inverse von e bezüglich $\varphi(N)$:

$$e \cdot d \equiv 1 \ (mod\ \varphi(N))$$

Daraus lassen sich die Schlüssel wie folgt bilden:

Der öffentliche Schlüssel:

$$(e, N)$$

und der private Schlüssel:

$$(d, N)$$

Die Werte von p, q und $\varphi(N)$ werden nicht mehr benötigt.

Eine Nachricht M wird mit dem öffentlichen Schlüssel verschlüsselt:

$$C \equiv M^e \ (mod\ N)$$

und der Chiffretext C wird mit dem privaten Schlüssel entschlüsselt:

$$M \equiv C^d \ (mod\ N)$$

4 Mögliche Angriffe auf RSA

Es gibt verschiedene Möglichkeiten mit RSA verschlüsselte Nachrichten zu brechen. Die bekannteste ist wahrscheinlich das Faktorisieren von N:

$$M \equiv C^{1/e\,(\,mod\ (p-1)(q-1))}\ (mod\ N)$$

Alternativ kann man mit Hilfe der Elementordnung von C in N RSA brechen:

$$M \equiv C^{1/e\,(\,mod\ order\ (C,N))}\ (mod\ N)$$

Diese beiden hier ausgewählten Verfahren werden im späteren Verlauf genauer erläutert.

5 Shor-Algorithmus

Der Shor-Algorithmus ist ein Faktorisierungsverfahren zur Anwendung auf Quantencomputern. Die Abbildung 1 gibt einen Überblick über den Shor-Algorithmus:

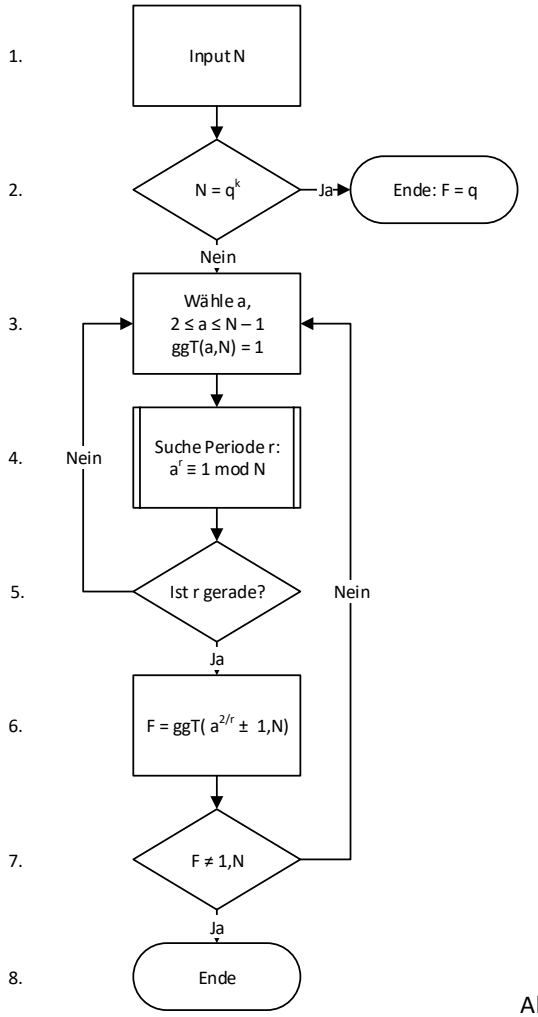

Abbildung 1: Shor Algorithmus

Im 1. Schritt wird die zu zerlegende Zahl N eingegeben.

Im 2. Schritt wird überprüft, ob N die Potenz einer Primzahl q mit Exponent k ist.

Im 3. Schritt wird ein zufällige Zahl a gewählt, für die gilt:

$$2 \leq a \leq N - 1$$

und

$$ggT(a, N) = 1$$

Sollte der $ggt(a, N) \neq 1$ sein, so ist damit ein Faktor gefunden und der Algorithmus terminiert.

Im 4. Schritt wird die Elementordnung r von a in N bestimmt.

$$a^r \equiv 1 \ (mod\ N)$$

Dies wird in Kapitel 7 detailliert beschrieben.

Im 5. Schritt wird überprüft, ob r gerade ist. Wenn ja, wird mit dem nächsten Schritt fortgefahren. Ansonsten wird zu Schritt 3 zurückgekehrt und mit einem neuen a gestartet.

Im 6. Schritt werden die zwei möglichen Faktoren von N bestimmt:

$$F_{\pm} = ggT\left(a^{r/2} \pm 1, N\right)$$

Im 7. Schritt wird überprüft, ob F_+ oder F_- nicht triviale Teiler von N sind:

$$F \neq 1, N$$

Wurde ein nicht trivialer Teiler gefunden, so wurde N erfolgreich faktorisiert. Ansonsten wird mit Schritt 3 und einem neuen a erneut gestartet.

Der größte Teil dieses Algorithmus kann von klassischen Computern effizient gelöst werden. Nur der 4. Schritt läuft in exponentieller Zeit. Deshalb kann der gesamte Algorithmus auf klassischen Computern nicht effizient berechnet werden. Ein Quantencomputer kann diese Aufgabe in polynomieller Zeit lösen. Damit kann auch N in polynomieller Zeit faktorisiert werden.

6 Quantencomputer

Ein klassischer Computer hat als kleinste Informationseinheit das Bit. Ein Bit kann die Werte 0 und 1 annehmen. Mithilfe von Transistoren können wir Operationen mit diesen Bits ausführen. Die kleinste Informationseinheit eines Quantencomputers ist ein Qubit (Quantenbit). Operationen werden mit speziellen Quantengattern ausgeführt.

6.1 Qubit

Ein Qubit kann durch verschiedene physikalische Methoden realisiert werden. Exemplarisch wird hier der Aufbau eines Qubits mit Hilfe eines Photons betrachtet. Die Information eines Qubit wird dabei repräsentiert durch die Ausrichtung der Polarisation. Ein Qubit kann so präpariert werden, dass es die klassischen Zustände 0 und 1 annehmen kann. Dies würde bei einem Photon einer horizontalen Polarisation (0) oder einer vertikalen Polarisation (1) entsprechen.

6.2 Superposition

Ein Qubit kann sich nun nicht nur in einem der beiden Basiszustände (Horizontal, Vertikal) befinden, sondern auch beliebige Werte dazwischen annehmen. Dieses Phänomen bezeichnet man als Superposition.
Ψ_1, Ψ_2 beschreiben die Vektoren der Basiszustände und Ψ ist der Vektor eines Photons in der Superposition (Abbildung 2).

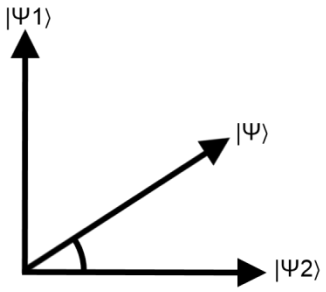

Abbildung 2: Superposition

Die Überlagerung von zwei Zuständen lässt sich mathematisch wie folgt beschreiben:

$$|\Psi\rangle = c_1|\Psi_1\rangle + c_2|\Psi_2\rangle\,;\; c_1, c_2 \in \mathbb{C}$$

Da laut Definition alle Vektoren normiert sein müssen, gilt:

$$|\Psi\rangle = \frac{1}{\sqrt{2}}(|\Psi_1\rangle + |\Psi_2\rangle)$$

Mit der Superposition lassen sich alle beliebigen Zustände gleichzeitig darstellen. Dieser physikalische Effekt ermöglicht die hohe Parallelität bei der Berechnung mit Quantencomputern. Bei einer Messung eines Qubits in der Superposition fällt es in einen der beiden Basiszustände zurück. Somit führt eine Messung immer zur Manipulation der Daten.

6.3 Verschränkung

Mehrere Qubits können verschränkt werden zu Quantenregistern. Auf das Phänomen der Verschränkung wird im Rahmen dieser Arbeit nicht weiter eingegangen.

6.4 Vorteile von Quantenrechnern

Der große Vorteil bei Quantenrechnern liegt in der Superposition. Ein klassisches 4 Bit Register kann $2^4 = 16$ Zustände darstellen. Es kann aber nur maximal einer dieser 16 Werte zurzeit gespeichert werden. Bei einem 4 Qubit Register in der Superposition können alle 16 Zustände gleichzeitig dargestellt werden. Zusätzlich werden alle Rechenoperationen, die auf dem Register ausgeführt werden, auf allen 16 Werten parallel berechnet. Bei der Messung der Quantenregister begeben sich die einzelnen Qubits zufällig in einen ihrer Basiszustände.

6.5 Was Quantenrechner können und was nicht

Quantencomputer können alle Probleme lösen, die ein klassischer Computer lösen kann. Probleme, die unlösbar für einen klassischen Computer sind, können auch nicht von Quantencomputern gelöst werden. Ein Beispiel hierfür ist das Wortproblem der Gruppentheorie.

7 Bestimmung der Periode

Um auf einem klassischen Computer die Periode eines Elements zu bestimmen, muss die gesuchte Zahl so oft mit sich selbst multipliziert werden, bis der Rest 1 ergibt.

Exemplarisch wollen wir die Periode von x in N berechnen, mit $x = 3; N = 156790$:

3^1	mod	156790	$=$	3
3^2	mod	156790	$=$	9
			$\vdots$	
3^{10451}	mod	156790	$=$	117666
3^{10452}	mod	156790	$=$	1

Somit muss man die 3 10452-mal mit sich selbst multiplizieren, um die Periode zu bestimmen.

Wenn die Ordnung in N mit 200 Stellen bestimmt werden soll, können mehr als 10^{150} Multiplikationen benötigt werden. Dies würde auf einem modernen Supercomputer bis zu 10^{80} Jahre dauern. Auf Quantencomputern hingegen könnte dieses Problem in absehbarer Zeit gelöst werden. Im Folgenden wird ein Algorithmus zur Bestimmung der Periode auf einem Quantencomputer beschrieben.

Als erstes muss die Größe des benötigten Registers bestimmt werden. Dabei ist s eine gewählte Zahl, die eine Zweierpotenz sein muss, für die gilt:

$$N^2 < s < 2N^2$$

Daraus ergibt sich die Größe des Quantenregisters wie folgt:

$$s = 2^t$$

Dabei entspricht t der Anzahl der benötigten Qubits für das Quantenregister.
Für die Berechnung werden zwei Quantenregister benötigt. Die Zustände der beiden Quantenregister werden mit $|\Psi_n\rangle$ benannt. Die einzelnen Register werden in der Bra-Ket-Notation dargestellt.

$$|\Psi_0\rangle = |\alpha\rangle|\beta\rangle$$

Im ersten Schritt werden alle Qubits des ersten Registers in die Superposition gebracht. Das zweite Register wird mit 0 initialisiert.

$$|\Psi_1\rangle = \frac{1}{\sqrt{s}} \sum_{x=0}^{s-1} |\mathrm{x}\rangle\,|0\rangle$$

Im ersten Register sind nun durch die Superposition alle Zahlen von 0 bis $s-1$ mit gleicher Wahrscheinlichkeit vorhanden. Jedes einzelne Qubit wird durch folgende Formel beschrieben:

$$\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$$

Im nächsten Schritt werden im zweiten Register $C^x \ (mod\ N)$ mit allen Werten aus dem ersten Register berechnet. Dieser Vorgang wird parallel für alle x ausgeführt.

$$|\Psi_2\rangle = \frac{1}{\sqrt{s}} \sum_{x=0}^{s-1} |\mathrm{x}\rangle\,|C^x \ (mod\ N)\rangle$$

Nach der Berechnung liegt im zweiten Register ein periodisches Muster vor. Auf dieses Muster wird die Quanten-Fouriertransformation angewendet. Bis zu diesem Zeitpunkt befinden sich die Register in der Superposition. Durch Messen gelangen die Qubits aus der Superposition in einen der Basiszustände, der mit einer bestimmten Wahrscheinlichkeit die Periode r ergibt. Wie hoch die Wahrscheinlichkeit für das Lesen der Periode ist, hängt von vielen Faktoren ab, auf die hier nicht weiter eingegangen wird. Sollte das Ergebnis fehlerhaft sein, so muss der gesamte Vorgang wiederholt werden.

8 Beispielrechnung Shor-Algorithmus

Im Weiteren soll der Shor-Algorithmus an zwei Beispielen verdeutlicht werden. Im ersten Beispiel wird der Algorithmus vollständig ausgeführt. Im zweiten Beispiel werden einige weitere Zahlenkonstellationen erläutert.

8.1 Beispiel für einen erfolgreichen Shor-Algorithmus

In diesem Beispiel wird die Zahl 15 faktorisiert. Daraus folgt:

$$N = 15$$

Im 2. Schritt wird überprüft, ob es ein q, k gibt für das gilt:

$$N = q^k$$

Dafür wird berechnet $\sqrt[n]{N}$, mit $n = 2$ bis $\sqrt[n+i]{N} < 2$:

$\sqrt[2]{15}$	3,873
$\sqrt[3]{15}$	2,466
$\sqrt[4]{15}$	1,968

Es wird mit Schritt 3 fortgefahren:

Es wird ein a gewählt, für das gelten muss:

$$2 \leq a \leq N - 1$$
$$ggT(a,N) = 1$$

In diesem Beispiel ist $a = 7$, mit:

$$2 \leq 7 \leq N - 1$$
$$ggT(7,15) = 1$$

Im folgenden Schritt wird die Periode bestimmt.

Zuerst wird s und die Registergröße t bestimmt:

$$s = 2^t : N^2 < s < 2N^2$$
$$15^2 < s < 2 \cdot 15^2; 225 < s < 450$$
$$s = 2^8 = 256$$
$$t = 8$$

Für die folgende Berechnung sind zwei Quantenregister mit jeweils mindestens 8 Qubit notwendig. Die Größe von s und t ist für die weitere Berechnung nicht relevant.

Die Quantenregister werden initialisiert:

$$|\Psi_1\rangle = \frac{1}{\sqrt{s}} \sum_{x=0}^{s-1} |\mathrm{x}\rangle\,|0\rangle$$

Nach der Initialisierung sehen die Quantenregister wie folgt aus:

$$|\Psi_1\rangle = \frac{1}{\sqrt{s}} \begin{bmatrix} |0\rangle|0\rangle + |1\rangle|0\rangle + |2\rangle|0\rangle + |3\rangle|0\rangle + \\ |4\rangle|0\rangle + |5\rangle|0\rangle + |6\rangle|0\rangle + |7\rangle|0\rangle + \\ |8\rangle|0\rangle + |9\rangle|0\rangle + |10\rangle|0\rangle + |11\rangle|0\rangle + \\ \cdots \end{bmatrix}$$

Es können nun die Werte des zweiten Registers berechnet werden:

$$|\Psi_1\rangle \rightarrow |\Psi_2\rangle = \frac{1}{\sqrt{s}} \sum_{x=0}^{s-1} |\mathrm{x}\rangle\,|a^x \ (mod\ N)\rangle$$

Es ergibt sich folgender Zustand:

$$|\Psi_2\rangle = \frac{1}{\sqrt{s}} \begin{bmatrix} |0\rangle|1\rangle + |1\rangle|7\rangle + |2\rangle|4\rangle + |3\rangle|13\rangle + \\ |4\rangle|1\rangle + |5\rangle|7\rangle + |6\rangle|4\rangle + |7\rangle|13\rangle + \\ |8\rangle|1\rangle + |9\rangle|7\rangle + |10\rangle|4\rangle + |11\rangle|13\rangle + \\ \cdots \end{bmatrix}$$

Im zweiten Register zeigt sich nun ein Periodisches Muster, auf das die Quanten-Fouriertransformation angewendet wird. Es erfolgt die Messung des Quantenregisters.

In diesem Beispiel ist $r = 4$.

Es wird getestet:

$$a^r \equiv 1\ (mod\ N)$$

$$7^4 \equiv 1\ (mod\ 15)$$

$$2401 \equiv 1 (mod\ 15)$$

Im 5. Schritt wird überprüft, ob r gerade ist:

$$r = n \cdot 2$$

$$r = 4 = 2 \cdot 2$$

Es werden die Faktoren von N berechnet:

$$F_+ = ggT\left(a^{r/2} + 1, N\right);\ F_- = ggT\left(a^{r/2} - 1, N\right)$$

$$F_+ = ggT\left(7^{4/2} + 1{,}15\right) = ggT(50{,}15) = 5$$

$$F_- = ggT\left(7^{4/2} - 1{,}15\right) = ggT(48{,}15) = 3$$

Da

$$5 \neq 1; 5 \neq 15$$

und

$$3 \neq 1; 3 \neq 15$$

sind, sind 3 und 5 nicht triviale Teiler von 15:

$$F = \{3{,}5\}$$

8.2 Beispiele für fehlerhaften Shor-Algorithmus

In den nächsten Beispielen werden auch Konstellationen gezeigt, bei denen der Shor-Algorithmus fehlschlägt.

Die zu faktorisierende Zahl N ist bei den folgenden Beispielen 21. Getestet werden soll $a = 2, 4, 5$.

Bei $a = 2$ ist die Periode $r = 6$. Daraus werden die Faktoren berechnet:

$$F_+ = ggT(7{,}21) = 3; F_- = ggT(9{,}21) = 3$$

$$F = 3$$

Bei $a = 4$ ist die Periode $r = 3$. An diesem Punkt bricht der Shor-Algorithmus ab, da die Periode gerade sein muss.

Bei $a = 5$ ist die Periode $r = 6$. Daraus werden die Faktoren berechnet:

$$r = 6 \rightarrow F_+ = ggT(126{,}21) = 21; F_- = ggT(124{,}21) = 1$$

Beide Faktoren sind trivial. Auch hier muss der Algorithmus mit einem neuen a gestartet werden.

8.3 Wahrscheinlichkeit des Shor-Algorithmus

Wie in den Beispielen zu sehen, ist der Shor-Algorithmus nicht immer erfolgreich. Die Wahrscheinlichkeit für eine erfolgreiche Durchführung wird wie folgt beschrieben:

$$W \geq 1 - \frac{1}{2^{k-1}}$$

wobei k der Anzahl der verschiedenen Primfaktoren in N entspricht und $k > 1$ gelten muss, da ansonsten die Wahrscheinlichkeit $W = 0$ eintritt.

Beispielhaft wird die Wahrscheinlichkeit für $N = 21$ berechnet:

$$N = 21 = 3 \cdot 7$$

$$k = |\{3,7\}| = 2$$

$$W \geq 1 - \frac{1}{2^{2-1}}$$

$$W \geq 0{,}5$$

Bei 2 Faktoren ergibt sich eine Wahrscheinlichkeit von 50% einen nicht trivialen Faktor zu finden.

Nach t Durchläufen beträgt die Fehlerwahrscheinlichkeit:

$$W = 2^{-t}$$

So ergibt sich nach 10 Durchläufen eine Fehlerwahrscheinlichkeit von:

$$W = 2^{-10} = 0{,}0009765625 \sim 0{,}1\%$$

und nach 20 Durchläufen:

$$W = 2^{-20} = 0{,}00000095367431640625 \sim 0{,}0001\%$$

9 Vorgehensweise zum Brechen von RSA

Im Folgenden wird ein RSA Schlüsselpaar erzeugt, wie in Kapitel 3 beschrieben:

$$p = 7; q = 11$$

$$N = p \cdot q = 77$$

$$\varphi(N) = (p-1)(q-1) = 60$$

$$e = 47$$

$$e \cdot d \equiv 1 \ (mod\ \varphi(N))$$

$$d = 23$$

Der öffentliche Schlüssel:

$$e = 47;\ N = 77$$

und der private Schlüssel:

$$d = 23;\ N = 77$$

Im Beispiel wird die Nachricht $M = 3$ verschlüsselt:

$$C \equiv M^e \ (mod\ N)$$

$$C = 75$$

Dem Angreifer stehen nur der Chiffretext C und der öffentliche Schlüssel (e, N) zur Verfügung.

9.1 Elementordnung

In diesem Beispiel wird unter Verwendung der Elementordnung von C in N RSA gebrochen:

$$M \equiv C^{1/e\,(\,mod\ order\ (C,N))}\ (mod\ N)$$

Als erstes wird die Ordnung von C in N berechnet. Hierbei sollte ein Quantencomputer zum Einsatz kommen.

$$order(C,N) \rightarrow C^r \equiv 1\ (mod\ N)$$

$$r = 30$$

Mit der Ordnung r kann nun die Nachricht M berechnet werden.

$$M \equiv 75^{1/47\,(\,mod\ 30)}\ (mod\ 77)$$

$$M = 3$$

9.2 Shor-Algorithmus

In diesem Beispiel wird N mit Hilfe des Shor-Algorithmus faktorisiert und so die Nachricht C entschlüsselt. Auch hier wird ein Quantencomputer für eine effiziente Lösung vorausgesetzt.

$$M \equiv C^{1/e\,(\,mod\ (p-1)(q-1))}\ (mod\ N)$$

Es wird ein zufälliges gültiges a gewählt:

$$a = 3$$

Es wird die Elementordnung von a in N berechnet:

$$r = order(3{,}77) = 30$$

Da r gerade ist, werden nun die Faktoren berechnet:

$$F_{\pm} = ggt\left(a^{r/2} \pm 1, N\right)$$

$$F_{+} = ggT(3^{15} + 1{,}77) = 7; F_{-} = ggT(3^{15} - 1{,}77) = 11$$

$$p = 7; q = 11$$

Da die Faktoren bekannt sind, kann C entschlüsselt werden:

$$M \equiv 75^{1/47\,(\,mod\ (7-1)(11-1))}\ (mod\ 77)$$

$$M = 3$$

10 Komplexitätsanalyse

Das Brechen von RSA auf klassischen Computern ist bis jetzt nicht in polynomineller Zeit möglich. Auf Quantenrechner hingegen kann RSA in polynomineller Zeit gebrochen werden. Die Laufzeitabschätzung ist:

$$\mathcal{O}((\log N)^3)$$

Es ist anzumerken, dass nicht bewiesen ist, ob das Faktorisierungsproblem NP-vollständig ist. Es gibt auch noch kein NP-vollständiges Problem, das auf einem Quantencomputer gelöst werden kann.

11 Fazit

RSA stellt zurzeit ein sehr sicheres Verschlüsselungsverfahren dar, das in weiten Bereichen des Internet angewendet werden kann. Alle Verfahren RSA zu brechen, benötigen auf klassischen Systemen exponentielle Zeit und stellen momentan keine Gefahren für RSA dar. Mit Quantencomputern könnte RSA in polynomieller Zeit gebrochen werden. Auch das Diffie-Hellmann-Verfahren könnte dann in polynomieller Zeit gebrochen werden, da es mit ähnlichen mathematischen Grundsätzen arbeitet. Bis jetzt ist nicht bekannt, ob es bereits geeignete Quantencomputer gibt, oder wann sie zur Verfügung stehen könnten. Schätzungen dazu geben einen Zeitraum von 20 bis 50 Jahren an.

12 Quellenverzeichnis

12.1 Literatur

- Yan, Song Y (2007): *Cryptanalytic attacks on RSA*. New York: Springer
- Audretsch, Jürgen (2002): *Verschränkte Welt*. Weinheim: Wiley-VCH

12.2 Internet

Letzter Zugriff am 5.6.2016

- http://www.quantencomputer.de/
- http://www.weltderphysik.de/gebiet/technik/quanten-technik/einfuehrung-quantencomputer/
- http://onlinelibrary.wiley.com/doi/10.1002/phbl.19990550909/pdf
- http://www.libquantum.de/files/libquantum.pdf
- http://www.physik.uni-siegen.de/quantenoptik/lehre/hauptseminarqm/qft3.pdf