PGP

Pretty Good Privacy

RSA–Verschlüsselung für Jedermann

von Philip Zimmermann

übersetzt von Gunnar Schmidt
gekürzte Tex–Version von Lars Oergel

27. Oktober 1999

Inhaltsverzeichnis

1 Übersicht

PGP(tm) benutzt RSA–Verschlüselung um E–Mail und Daten zu schützen. Sie können mit Menschen geheime Daten austauschen, die Sie noch nie getroffen haben und brauchen keine „sicheren“ Kanäle um Paßworte auszutauschen. PGP hat viele Funktionen, ist schnell, hat ein gutes Schlüsselmanagement, Datenkompression und gute Ergonomie.

2 Schnellübersicht

Pretty Good(tm) Privacy (PGP), von Phil's Pretty Good Software ist eine hochsichere kryptografische Software für MSDOS, Unix, VAX/VMS und andere Computer. PGP erlaubt es, geheime Dateien oder Nachrichten bequem und mit Authentizität auszutauschen. Geheim bedeutet, nur die Personen, die eine Nachricht lesen sollen, können sie auch lesen. Authentizität bedeutet, wenn eine Nachricht von einer bestimmten Person zu kommen scheint, kann sie nur von dieser Person kommen. Bequem heißt, Sicherheit und Authentizität ohne die Schwierigkeiten wie z.B. der Schlüsselverwaltung herkömmlicher kryptografischer Programme. Es werden keine „sicheren“ Kanäle benötigt, um die Schlüssel auszutauschen, und dies macht PGP wesentlich einfacher in der Benutzung. Dies kommt daher, daß PGP auf einer neuen Technologie basiert, die RSA–Verschlüsselung heißt.

PGP kombiniert die Bequemlichkeit des Rivest–Shamir–Adleman (RSA) Kryptosystems mit der Geschwindigkeit herkömmlicher Verschlüsselungsmethoden, digitalen Unterschriften, Datenkompression vor der Verschlüsselung, gute Ergonomie und hervorragende Schlüsselverwaltung. Und PGP führt die RSA–Funktionen schneller durch als die meisten anderen Software–Lösungen. PGP ist RSA–Verschlüsselung für Jedermann.

Dieses Dokument, „Teil I: Die wichtigsten Themen“, erklärt nur die wichtigsten PGP–Möglichkeiten und sollte von allen PGP–Benutzern gelesen werden. „Teil II: Spezielle Themen“ handelt von den besonderen Fähigkeiten von PGP und sollte von den ernsthafteren Anwendern gelesen werden. Keines der beiden Teile erklärt die zugrundeliegenden technologischen Details kryptografischer Algorithmen oder Datenstrukturen.

3 Warum brauchen Sie PGP?

Es ist persönlich. Es ist privat. Und es geht niemanden außer Sie etwas an. Vielleicht planen Sie eine politische Kampagne, Ihre Steuern oder eine unerlaubte Affäre. Oder vielleicht tun Sie etwas, von dem Sie denken, daß es nicht illegal sein sollte, aber es ist. Was auch immer es ist, Sie wollen nicht, daß Ihre private elektronische Post (E–Mail) oder Ihre vertraulichen Dokumente von irgend jemand anders gelesen werden. Es ist nichts falsch daran, Privatsphäre zu verlangen.

Vielleicht denken Sie, Ihre E–Mail ist legitim genug um ohne Verschlüsselung auszukommen. Wenn Sie ein so gesetzestreuer Bürger sind, der nichts zu verstecken hat, warum schreiben Sie Ihre Briefe nicht auf Postkarten? Warum verweigern Sie einen Drogentest, wenn er verlangt wird? Warum verlangen Sie einen Durchsuchungsbefehl wenn die Polizei vor der Tür steht? Versuchen Sie, etwas zu verstecken? Sie

müssen ein Drogendealer sein, wenn Sie Ihre Post in Umschlägen verstecken. Oder vielleicht einfach nur eine paranoide Nuß? Müssen gesetzestreue Bürger Ihre E–Mail verschlüsseln?

Was, wenn jeder glauben würde, gesetzestreue Bürger müßten Postkarten für Ihre Post benutzen? Wenn irgendjemand seine Privatsphäre gelten machen und seine Post in Umschläge stecken würde, würde das sofort Verdacht erregen. Vielleicht würden die Behörden seine Umschläge öffnen um zu sehen, was er versteckt. Glücklicherweise leben wir nicht in einer solchen Welt, weil jeder seine meiste Post mit Umschlägen schützt. So erregt niemand Verdacht, wenn er seine Privatsphäre mit Umschlägen geltend macht. Nummern sind Sicherheit. Analog dazu wäre es schön, wenn jeder, unschuldig oder nicht, seine E–Mail verschlüsseln würde, denn dann würde niemand Verdacht erregen, wenn er seine private E–Mail mit (Verschlüsselungs–) Umschlägen schützt. Sehen Sie es als eine Form der Solidarität.

Wenn die Regierung heute die Privatsphäre von normalen Leuten verletzen will, muß sie eine ganze Menge Geld ausgeben und Aufwand treiben um die Post abzufangen, mit Dampf zu öffnen und zu lesen, oder um Telefongespräche mitzuhören oder aufzuzeichnen. Diese Art aufwandsintensive Beobachtung ist im großen Stile unpraktikabel. Sie wird nur in wichtigen Fällen betrieben, wenn es lohnend erscheint.

Mehr und mehr unserer privaten Kommunikation geht durch elektronische Kanäle. E–Mail ersetzt immer mehr die Briefpost. Aber E–Mail–Nachrichten sind zu einfach abzufangen und nach interessanten Stichwörtern zu durchsuchen. Das kann im großen Stile sehr einfach, routinemäßig, automatisch und unerkennbar gemacht werden. Internationale Kabelnetze werden jetzt schon von der NSA (National Security Agency/USA–Behörde – Anm.d.Ü.) im großen Stile gescannt.

Wir gehen auf eine Zukunft zu, in der die Welt überzogen sein wird von hochkapazitiven Fiber–optischen Netzwerken, die unsere allgegenwärtigen Personalcomputer verbinden. E–Mail wird die Norm sein, nicht die Neuigkeit, die sie heute ist. Die Regierung wird unsere E–Mail mit Verschlüsselungsprotokollen sichern, die von der Regierung geschaffen wurden. Die meisten Menschen werden damit zufrieden sein. Aber vielleicht werden einige ihre eigenen Sicherheitsmaßstäbe setzen.

Die Senatsverordnung 266, eine 1991er Anti–Verbrechens–Erklärung beinhaltete eine sehr störende Resolution. Wenn diese nicht bindende Erklärung Realität geworden wäre, wären Hersteller von sicheren Kommunikationsmitteln dazu verpflichtet gewesen, spezielle „Falltüren“ in ihre Produkte einzubauen, damit die Regierung jedermannes verschlüsselte Nachrichten lesen kann. Sie lautete: „Es liegt im Sinne des Kongresses, das Anbieter von elektronischen Kommunikationsservices und Hersteller von elektronischen Kommunikationsmitteln sicherstellen müssen, daß die Regierung unverschlüsselte Inhalte von Sprache, Daten und andere Kommunikationswegen erhalten kann, wenn das Gesetz es erfordert.“ Diese Maßnahme wurde aber nach rigorosen Protesten von ziviler und industrieller Seite nicht Gesetz.

1992 wurde das „FBI Digital Telephony wiretap proposal“ im Kongreß vorgestellt. Alle Hersteller von Kommunikations–Equipment sollten spezielle „Abhörports“ in ihre Geräte einbauen, was es dem FBI möglich machen sollte, alle Formen elektronischer Kommunikation vom Büro aus abhören zu können. Obwohl es 1992 wegen Ablehnung durch die Bürger niemals Unterstützung fand wurde es 1994 wieder vorgeschlagen.

Das größte Alarmzeichen ist aber die neue Verschlüsselungs–Politik des Weißen Hauses, die schon seit dem Beginn der Bush–Zeit von der NSA entwickelt und am 16.

April 1993 bekannt wurde. Das Herzstück dieser Politik ist ein von der Regierung gebautes Verschlüsselungs–Gerät, der „Clipper"–Chip, welcher einen neuen, von der NSA entwickelten geheimen Verschlüsselungsalgorithmus trägt. Die Regierung fordert die Privatindustrie auf, den Chip in ihre sicheren Kommunikationsgeräte einzubauen, wie verschlüsseltes Telefon, FAX, usw. AT&T baut den Clipper schon in seine „secure voice"–Produkte ein. Das Problem: In der Produktion bekommt jeder Chip seinen einzigartigen Schlüssel und die Regierung erhält eine Kopie dieses Schlüssels. Keine Problem soweit, die Regierung verspricht, den Schlüssel nur zu benutzen um Ihre geheimen Nachrichten zu lesen, wenn sie durch das Gesetz dazu autorisiert wird. Aber natürlich, um Clipper erst effektiv zu machen wäre der nächste logische Schritt, alle anderen Formen der Kryptografie zu illegalisieren.

Wenn Privatsphäre illegal ist, werden nur Illegale Privatsphäre haben. Geheimdienste haben Zugang zu guter kryptografischer Technologie. Waffenhändler und Drogendealer ebenfalls. Ölfirmen und andere große Konzerne ebenfalls. Nur die normalen Leute und kleine politische Organisationen hatten bis jetzt keinen Zugriff auf kryptografische Technologie mit „militärischer Sicherheit". Bis jetzt. PGP fordert die Menschen dazu auf, ihre Privatsphäre in die eigenen Hände zu nehmen. Dafür ist ein wachsender sozialer Bedarf vorhanden. Deswegen habe ich es geschrieben.

4 Wie es funktioniert

Es wäre hilfreich, wenn Sie schon etwas mit dem Konzept der Verschlüsselung im allgemeinen und mit der RSA–Verschlüsselung im Besonderen vertraut wären. Trotzdem hier ein paar einführende Worte über RSA–Verschlüsselung.

Zuerst einige elementare Terminologien. Nehmen wir an, ich will Ihnen eine Nachricht schicken, aber ich will, das niemand außer Ihnen sie lesen kann. Ich kann die Nachricht „verschlüsseln", was bedeutet, ich zerwürfle sie auf eine hoffnungslos komplizierte Art, so daß niemand außer Ihnen in der Lage ist, sie zu lesen. Ich benutze einen kryptografischen „Schlüssel" um die Nachricht zu verschlüsseln und Sie müssen denselben Schlüssel verwenden um die Nachricht wieder zu entschlüsseln. So funktioniert es jedenfalls auf die herkömmliche Weise in „Ein–Schlüssel"–Kryptosystemen.

In herkömmlichen Kryptosystemen, wie z.B. dem „US Federal Data Encryption Standard (DES)" wird ein einzelner Schlüssel für Ver- und Entschlüsselung benutzt. Das bedeutet, der Schlüssel muß zuerst über sichere Kanäle übertragen werden, so daß beide Seiten ihn kennen, bevor verschlüsselte Nachrichten über „unsichere" Kanäle übertragen werden können. Das hört sich sinnlos an. Wenn ich einen sicheren Kanal habe, um den Schlüssel zu übertragen, wozu brauche ich dann Verschlüsselung?

In RSA–Kryptosystemen hat jeder zwei Schlüssel, die zueinander komplementär sind, einen öffentlich bekannten und einen geheimen (auch oft Privatschlüssel genannt). Jeder Schlüssel öffnet den Code, den der andere erzeugt. Den öffentlichen Schlüssel zu kennen hilft Ihnen nicht, den zugehörigen geheimen zu erzeugen. Der öffentliche Schlössel kann weit über ein Kommunikations–Netz verbreitet sein. Dieses Protokoll erzeugt Privatsphäre ohne die sicheren Kanäle zu brauchen, die für konventionelle Kryptosysteme benötigt werden.

Jeder kann den öffentlichen Schlüssel des Empfängers benutzen, um eine Nachricht für ihn zu verschlüsseln, und dieser benutzt seinen geheimen, um sie wieder zu

entschlüsseln. Niemand außer ihm kann die Nachricht entschlüsseln, weil niemand außer ihm Zugriff auf den geheimen Schlüssel hat. Nicht einmal die Person, welche die Nachricht verschlüsselt hat, kann sie entschlüsseln.

Authentifikation einer Nachricht ist auch möglich. Der geheime Schlüssel des Absenders kann benutzt werden, um eine Nachricht zu verschlüsseln, sie zu „unterschreiben". Das erzeugt eine digitale Unterschrift der Nachricht, welche der Empfänger (oder jeder andere) überprüfen kann, indem er den öffentlichen Schlüssel benutzt, um sie zu entschlüsseln. Dieses beweist, daß der Absender wirklich der wahre Erzeuger der Nachricht ist und daß die Nachricht nicht von jemand anderem verändert wurde, weil nur der Absender den geheimen Schlüssel besitzt, der die Unterschrift erzeugte. Eine Fälschung der Unterschrift ist unmöglich und der Absender kann seine Unterschrift später nicht leugnen.

Diese beiden Prozesse können kombiniert werden, um Geheimhaltung und Authentizität zu gewährleisten, indem Sie zuerst Ihre Nachricht mit Ihrem geheimen Schlüssel unterschreiben und dann mit dem öffentlichen Schlüssel des Empfängers verschlüsseln. Der Empfänger kehrt dieses Schema um, indem er zuerst die Nachricht mit seinem geheimen Schlüssel entschlüsselt und dann Ihre Unterschrift mit Ihrem öffentlichen Schlüssel prüft. Diese Schritte werden vom Programm automatisch ausgeführt.

Weil die RSA–Verschlüsselung wesentlich langsamer ist als konventionelle Methoden, ist es besser, ein schnelleres, hochsicheres konventionelles Verfahren zu verwenden, um den Text zu verschlüsseln. Dieser originale unverschlüsselte Text wird „einfacher Text" genannt. In einem für den Benutzer unsichtbarem Prozeß wird zuerst ein einmaliger Zufallsschlüssel erzeugt, mit dem der Text dann auf konventionelle Weise verschlüsselt wird. Der Zufallsschlüssel wird jeweils nur ein einziges Mal verwendet. Dann wird dieser Zufallsschlüssel mit dem öffentlichen Schlüssel des Empfängers verschlüsselt und mit der Nachricht zu ihm geschickt. Er entschlüsselt ihn mit seinem geheimen Schlüssel und benutzt ihn, um Ihre Nachricht wieder mit den schnellen konventionellen Methoden zu entschlüsseln.

Öffentliche Schlüssel werden in individuellen „Schlüssel–Zertifikaten" aufbewahrt, welche die „User ID" (der Name des Benutzers), die Entstehungszeit des Schlüssels und das eigentliche „Schlüsselmaterial" enthalten. Öffentliche Schlüsselzertifikate enthalten öffentliche Schlüssel, während geheime Schlüsselzertifikate geheime Schlüssel enthalten. Jeder geheime Schlüssel ist nochmal mit einem Paßwort verschlüsselt, um ihn zu schützen, sollte er gestohlen werden. Eine Schlüsseldatei, oder „Schlüsselbund" enthält ein oder mehrere dieser Schlüsselzertifikate. Öffentliche und geheime Schlüssel werden in getrennten Schlüsselbunden aufbewahrt.

Die Schlüssel werden intern auch mit einer „Key ID" referenziert, welche eine „Abkürzung" (die untersten 64 Bit) des öffentlichen Schlüssels ist. Wenn diese Key ID angezeigt wird, sehen sie aber nur 32 Bit, um noch weiter abzukürzen. Viele Schlüssel können die gleiche User ID haben, aber in der Praxis haben keine zwei Schlüssel die gleiche Key ID.

PGP benutzt „Nachrichten–Extrakte" um Unterschriften zu erzeugen. Ein Nachrichtenextrakt ist eine 128 Bit lange kryptografisch sehr starke Hash–Funktion. Es funktioniert etwa wie eine Checksumme oder CRC, indem es die Nachricht in kompakter Form „repräsentiert" und dazu benutzt wird, um Veränderungen in der Nachricht zu entdecken. Aber im Gegensatz zu einer CRC ist es für einen Angreifer praktisch unmöglich, eine Nachricht zu erzeugen, aus der das gleiche Nachrichtenextrakt entsteht. Dieser Nachrichtenextrakt wird mit dem geheimen Schlüssel des Absenders verschlüsselt um eine digitale Unterschrift zu erzeugen.

Dokumente werden unterschrieben, indem ihnen ein Unterschriftszertifikat vorangestellt wird, welches die Key ID des Schlüssels enthält, mit welchem sie unterschrieben wurde, das verschlüsselte Nachrichtenextrakt und eine Zeitmarke, mit der festgestellt werden kann, wann die Unterschrift erzeugt wurde. Die Key ID wird vom Empfänger benutzt, um den zum Überprüfen nötigen öffentlichen Schlüssel zu finden. Das Programm des Empfängers sucht automatisch die zur Key ID passende User ID und den Schlüssel im öffentlichen Schlüsselbund.

Verschlüsselten Dateien wird die Key ID des öffentlichen Schlüssels vorangestellt, der benutzt wurde, um sie zu verschlüsseln. Das Programm des Empfängers sucht automatisch den zur Entschlüsselung nötigen Schlüssel im geheimen Schlüsselbund.

Diese zwei Schlüsselbunde sind die prinzipielle Methode, um öffentliche und geheime Schlüssel zu verwalten. Anstatt jeden einzelnen Schlüssel in einzelnen Datei zu verwalten, werden sie in Schlüsselbunden gesammelt, um ihr Auffinden durch User ID und Key ID möglich zu machen. Jeder Nutzer erhält sein eigenes Paar Schlüsselbunde. Erzeugte öffentliche Schlüssel werden temporär in einzelnen Dateien gehalten, damit Sie sie zu Ihren Freunden senden können, die sie dann zu ihren öffentlichen Schlüsselbunden hinzufügen können.

5 Wie Sie PGP benutzen

5.1 Um eine Benutzungszusammenfassung zu sehen

Um eine Kurzzusammenfassung der Befehle zu sehen, geben Sie ein:

```
pgp -h
```

5.2 Eine Nachricht verschlüsseln

Um eine Klartextdatei mit dem öffentlichen Schlüssel des Empfängers zu verschlüsseln, geben Sie ein:

```
pgp -e textdatei [seine User ID]
```

Dieses Kommando erzeugt ein Chiffrat mit dem Namen textdatei.pgp. Beispiele:

```
pgp -e brief.txt Alice
```

oder:

```
pgp -e brief.txt 'Alice S'
```

Das erste Beispiel sucht in Ihrem öffentlichen Schlüsselbund „pubring.pgp“ nach einem Schlüsselzertifikat, das die Zeichenkette „Alice“ im Feld User ID enthält. Das zweite Beispiel sucht nach „Alice S“. Sie können keine Leerzeichen verwenden, wenn Sie nicht die gesamte Zeichenkette in Anführungszeichen einschließen. Die Suche unterscheidet nicht zwischen Groß- und Kleinschreibung. Wenn PGP einen

passenden Schlüssel findet, benutzt es diesen, um „brief.txt“ zu verschlüsseln. Das Ergebnis ist eine Chiffrat–Datei mit dem Namen „brief.pgp“.

PGP versucht, den Klartext vor der Verschlüsselung zu komprimieren, was die Sicherheit gegenüber Kryptoanalyse wesentlich erhöht, und außerdem wird das Chiffrat meistens kleiner sein als das Original.

Wenn Sie die verschlüsselte Nachricht über E–Mail senden wollen, wandeln Sie es in druckbare ASCII–Zeichen („radix–64“–Format) um, indem Sie die Option `-a` anfügen, wie später erklärt wird.

5.3 Eine Nachricht für mehrere Empfänger verschlüsseln

Wenn Sie ein und dieselbe Nachricht an mehr als eine Person senden wollen, können Sie auch Verschlüsselung für mehrere Empfänger angeben, indem Sie einfach weitere User ID's hinzufügen, wie:

```
pgp -e brief.txt Alice Bob Carol
```

Dies würde ein Chiffrat namens „brief.pgp“ erzeugen, das von Alice, Bob oder Carol entschlüsselt werden kann. Die Anzahl der Empfänger ist beliebig.

5.4 Eine Nachricht unterschreiben

Um einen Klartext mit Ihrem geheimen Schlüssel zu unterschreiben, geben Sie ein:

```
pgp -s textdatei [-u ihre User ID]
```

Achtung, die [eckigen Klammern] weisen auf eine optionale Angabe hin, also geben Sie sie bitte nicht wirklich ein.

Dieser Befehl erzeugt eine unterschriebene Datei mit Namen textdatei.pgp.
Beispiel:

```
pgp -s brief.txt -u Bob
```

Dieser Befehl sucht in Ihrem geheimen Schlüsselbund „secring.pgp“ nach geheimen Schlüsselzertifikaten, welche die Zeichenkette „Bob“ irgendwo in der User ID enthalten. Sie heißen doch Bob, oder? Die Suche beachtet Groß- und Kleinschreibung nicht. Wenn der Schlüssel gefunden wird, benutzt PGP ihn, um die Datei „brief.txt“ zu unterschreiben und produziert eine Unterschriftsdatei „brief.pgp“.

Wenn Sie Ihre User ID nicht angeben, wird der erste Schlüssel in Ihrem geheimen Schlüsselbund verwendet.

PGP versucht, die Nachricht nach dem Unterschreiben zu komprimieren. Dadurch wird die unterschriebene Datei kleiner als die Originaldatei, was für Archivierungszwecke nützlich ist. Aber dadurch wird die Datei für den menschlichen Nutzer unlesbar, auch wenn die Originaldatei purer ASCII–Text war. Es wäre schön, wenn Sie eine unterschriebene Datei erzeugen könnten, die immer noch direkt lesbar ist. Das wäre teilweise nützlich, wenn sie die Datei per E–Mail verschicken wollen.

Um E–Mail Nachrichten zu unterschreiben, wenn das Resultat lesbar bleiben soll, ist es die bequemste Lösung, die CLEARSIG–Möglichkeit zu nutzen, die später erklärt wird. So wird die Unterschrift in druckbarer Form ans Ende des Textes gehängt und die Datenkomprimierung ausgeschaltet. Dadurch wird es möglich, daß der Empfänger die Datei lesen kann, sogar wenn er nicht PGP benutzt, um die Unterschrift zu prüfen. Diese Möglichkeit wird in der Sektion „CLEARSIG – Unterschriften verkapselt in lesbarem Text“ im „Teil II: Spezielle Funktionen“ erklärt. Wenn Sie nicht warten können, bis Sie diesen Abschnitt gelesen haben, können Sie sich anschauen, wie eine solchermaßen unterschriebene Datei aussehen würde, indem Sie eingeben:

```
pgp -sta nachricht.txt
```

Das erzeugt eine unterschriebene Nachrichtendatei mit dem Namen „nachricht.asc“ bestehend aus dem originalen Text, immer noch lesbar, und der Unterschrift als druckbarer ASCII–Text, fertig um sie per E–Mail zu senden. Dieses Beispiel geht davon aus, daß Sie die normalen Einstellungen für das CLEARSIG–Flag in der Konfigurationsdatei verwenden.

5.5 Unterschreiben und Verschlüsseln

Um einen Text mit Ihrem geheimen Schlüssel zu unterschreiben und ihn dann mit dem öffentlichen des Empfängers zu verschlüsseln, geben Sie ein:

```
pgp -es textdatei seine_UserID [-u ihre_userID]
```

Dieses Beispiel erzeugt eine unterschriebene Chiffratdatei textdatei.pgp. Ihr geheimer Schlüssel wird automatisch über ihre_userID auf Ihrem geheimen Schlüsselbund herausgesucht. Wenn Sie seine_UserID nicht angeben, wird PGP Sie danach fragen.

Wenn Sie ihre_userID frei lassen, wird Ihr erster geheimer Schlüssel verwendet.

Beachten Sie, daß PGP versucht, den Text zu komprimieren, bevor er verschlüsselt wird.

Wenn Sie diese Nachricht über E–Mail verschicken wollen, konvertieren sie die Ausgabe ins „Radix–64“–Format über die Option `-a`, wie weiter unten erklärt.

Sie können auch hier die Nachricht für mehrere Empfänger verschlüsseln, indem sie mehrere UserIDs angeben.

5.6 Konventionelle Verschlüsselung benutzen

Manchmal reicht es aus, Dateien mit der „alten“ Methode zu verschlüsseln, mit „Ein–Schlüssel“–Kryptografie. Das kann meistens dann nützlich sein, wenn Sie eine Datei nur verschlüsselt speichern wollen, aber nicht versenden. Weil dieselbe Person, welche die Daten verschlüsselt hat, sie auch wieder entschlüsselt, ist RSA–Verschlüsselung nicht unbedingt nötig.

Um eine Datei konventionell zu verschlüsseln, geben Sie ein:

```
pgp -c textdatei
```


Dieses Beispiel verschlüsselt die Datei textdatei und erzeugt ein Chiffrat textdatei.pgp ohne RSA–Verschlüsselung, Schlüsselbunde, User IDs oder ähnliches zu benutzen. PGP fragt Sie nach einem Paßwort, mit dem Sie Text verschlüsseln wollen. Dieses Paßwort muß nicht (besser: SOLLTE nicht) das gleiche sein, mit dem Sie Ihren geheimen RSA–Schlüssel schützen. Beachten Sie, daß PGP versucht, die Datei vor der Verschlüsselung zu komprimieren.

PGP wird niemals das gleiche Chiffrat aus einer Datei erzeugen, sogar wenn Sie das gleiche Paßwort immer wieder benutzen.

5.7 Entschlüsseln und Unterschriften prüfen

Um ein Chiffrat zu entschlüsseln oder eine Unterschriftsintegrität zu prüfen:

```
pgp chiffratdatei [-o dechiffredatei]
```

Wenn Sie zu der Chiffratdatei keine Extension angeben, wird .pgp angenommen. Der optionale Parameter `-o` gibt an, wo und unter welchem Namen die entschlüsselte Datei abgelegt wird. Wenn er nicht angegeben ist, benutzt PGP den Dateinamen des Chiffrats ohne Extension. Wenn eine Unterschrift im Text entdeckt wird, wird sie automatisch überprüft, und die volle User ID des Unterschreibenden wird angezeigt.

Beachten Sie, daß das „Auswickeln“ des Chiffrats vollautomatisch geschieht, egal ob die Datei verschlüsselt, unterschrieben oder beides ist. PGP benutzt das Key ID–Prefix um automatisch den passenden geheimen Schlüssel zur Entschlüsselung zu finden. Wenn sich eine Unterschrift im Text befindet, sucht PGP in Ihrem öffentlichen Schlüsselbund nach dem passendem Schlüssel für die Überprüfung. Wenn die richtigen Schlüssel in Ihren Schlüsselbunden vorhanden sind, sind keine Benutzereingaben mehr nötig, außer daß Sie eventuell nach dem Paßwort für Ihren geheimen Schlüssel gefragt werden. Wenn das Chiffrat konventionell verschlüsselt wurde, erkennt PGP dies und fragt Sie nach dem Paßwort.

5.8 Schlüsselverwaltung

Seit Cäsars Zeiten war das Schlüsselmanagement der schwierigste Teil der Kryptografie. Einer der wichtigsten Vorteile von PGP ist die superbe Schlüsselverwaltung.

5.8.1 RSA–Schlüssel erzeugen

Um Ihr eigenes einzigartiges öffentlich/geheimes Schlüsselpaar zu erzeugen:

```
pgp -kg
```

PGP zeigt Ihnen ein Menü mit bevorzugten Schlüssellängen (unterer kommerzieller Grad, hoher kommerzieller Grad, militärischer Grad) und fragt Sie, welche Schlüssellänge (bis mehr als tausend Bits) Sie wünschen. Je größer der Schlüssel, desto mehr Sicherheit haben Sie, aber der Preis, den Sie zahlen ist die Geschwindigkeit.

PGP fragt auch nach Ihrer User ID, was bedeutet: Ihr Name. Es ist eine gute Idee, Ihren vollen Namen für diese User ID zu benutzen, um das Risiko zu senken, daß Leute den falschen öffentlichen Schlüssel benutzen, um Nachrichten für Sie zu verschlüsseln. In der User ID sind Leerzeichen und Punktuation erlaubt. Es wäre hilfreich, wenn Sie Ihre E–Mail–Adresse in <Winkelklammern> hinter Ihrem Namen angeben, wie z.B.:

```
Robert M. Smith <rms@xyzcorp.com>
```

Wenn Sie keine E–Mail–Adresse haben, benutzen Sie Ihre Telefonnummer oder eine andere einmalige Information, um sicherzugehen, daß Ihre User ID einzigartig ist.

PGP fragt Sie auch nach Ihrem Sicherheitssatz, um den geheimen Schlüssel zu sichern, sollte er in die falschen Hände fallen. Niemand kann Ihren geheimen Schlüsselbund ohne den Sicherheitssatz benutzen. Der Sicherheitssatz ist wie ein Paßwort, aber er kann eine Wortgruppe oder Satz sein und alle Zeichen, Punkte, Leerzeichen und ähnliches enthalten, was Sie hineinschreiben wollen. Vergessen Sie diesen Satz nicht, es gibt KEINE Möglichkeit, ihn wieder zu ermitteln. Dieser Satz wird später immer dann benötigt, wenn Sie Ihren geheimen Schlüssel verwenden. Im Sicherheitssatz wird zwischen Groß- und Kleinschreibung unterschieden. Er sollte nicht zu kurz oder leicht zu erraten sein. Er wird niemals angezeigt und sie sollten ihn auch nirgends aufschreiben, wo ihn jemand anders sehen könnte oder ihn in Ihrem Computer speichern. Wenn Sie keinen Sicherheitssatz verwenden wollen (Sie Narr!), drücken Sie einfach Enter, wenn Sie nach dem Satz gefragt werden.

Das öffentlich/geheime Schlüsselpaar wird aus großen echten Zufallszahlen errechnet, die großteils dadurch erzeugt werden, indem PGP die Abstände zwischen Ihren Tastendrücken mit einem schnellen Timer mißt. Das Programm wird Sie anweisen, einigen Text einzugeben, um die Zufallszahlen zu erzeugen. Wenn Sie danach gefragt werden, sollten Sie einige Tasten in zufälligen Abständen drücken, und es wäre auch nicht schlecht, wenn der Text, den Sie eingeben auch einen zufälligen Inhalt hatte, weil ein Teil der „Zufälligkeit“ durch den Inhalt dessen, was Sie eingeben, erzeugt wird. Also drücken Sie bitte keine sich wiederholenden Tastenfolgen.

Beachten Sie, die Schlüsselerzeugung ist ein langwieriger Prozeß. Es könnte einige Sekunden dauern, um einen kleinen Schlüssel auf einem schnellen Prozessor zu erzeugen oder auch ein paar Minuten für einen langen Schlüssel auf einem IBM PC/XT. PGP wird seine Fortschritte während der Schlüsselerzeugung anzeigen.

Das erzeugte Schlüsselpaar wird in Ihren öffentlich/geheimen Schlüsselbunden abgelegt. Sie können Ihren neuen öffentlichen Schlüssel später mit dem Kommando **`-kx`** aus dem öffentlichen Schlüsselbund kopieren und in einer separaten Datei ablegen, die Sie dann Ihren Freunden geben können, damit sie ihn in ihre Schulüsselbunde ablegen können. Normalerweise behalten Sie Ihren geheimen Schlüssel in Ihrem geheimen Schlüsselbund. Jeder geheime Schlüssel wird durch seinen individuelle Sicherheitssatz geschützt.

Geben Sie Ihren geheimen Schlüssel niemandem. Aus demselben Grund sollten Sie auch nicht für Ihre Freunde Schlüsselpaare erzeugen. Jeder sollte sich sein eigenes Schlüsselpaar erzeugen. Behalten Sie immer die physikalische Kontrolle über Ihren geheimen Schlüssel und riskieren Sie nicht, daß er bekannt wird, weil Sie ihn auf einem entfernten PC (z.B. Netzserver) aufbewahren. Behalten Sie ihn in Ihrem eigenen PC (besser noch: auf Diskette).

Wenn PGP Ihnen mitteilt, es könne die PGP User's Guide nicht auf Ihrem Computer finden und sich weigert, ein Schlüsselpaar ohne dieses zu erzeugen, keine Panik.

Lesen Sie die Erklärung des NOMANUAL–Parameters in der Sektion „Konfigurationsparameter setzen“ im Spezialteil des PGP User's Guide.

5.8.2 Einen Schlüssel zu Ihrem Schlüsselbund hinzufügen

Manchmal werden Sie einen Schlüssel zu Ihrem Schlüsselbund hinzufügen wollen, der Ihnen von jemand anderem in der Form einer Schlüsseldatei gegeben wurde.

Um eine öffentliche/geheime Schlüsseldatei zu Ihrem Schlüsselbund hinzuzufügen: ([Klammern] bedeuten optionale Angabe)

```
pgp -ka Schluesseldatei [Schluesselbund]
```

Die Extension der Schlüsseldatei ist .pgp, wenn Sie diese nicht eingeben. Der Schlüsselbund ist entweder „pubring.pgp“ oder „secring.pgp“, abhängig davon, ob die Schlüsseldatei einen öffentlichen oder geheimen Schlüssel enthält. Sie können auch einen anderen Schlüsselbundnamen angeben, der wieder als Standardeinstellung die Extension .pgp hat.

Existiert der Schlüssel schon im Bund, wird PGP ihn nicht noch einmal hinzufügen. Alle Schlüssel der Schlüsseldatei werden hinzugefügt, außer den Duplikaten.

Später werden wir das Konzept erklären, Schlüssel mit Unterschriften zu versehen. Wenn ein Schlüssel der Datei Unterschriften enthält werden diese mit zum Bund hinzugefügt. Wenn sich der Schlüssel bereits im Bund befindet, überträgt PGP alle neuen Unterschriften aus der Schlüsseldatei.

PGP war ursprünglich dafür vorgesehen, kleine persönliche Schlüsselbunde zu verwalten. Wenn Sie wirklich große Schlüsselbunde verwalten wollen, lesen sie den Abschnitt „Verwalten großer öffentlicher Schlüsselbunde“ im Teil II dieser Dokumentation.

5.8.3 Einen Schlüssel aus dem Schlüsselbund entfernen

Um einen Schlüssel oder eine User ID aus Ihrem öffentlichen Schlüsselbund zu entfernen:

```
pgp -kr userid [Schluesselbund]
```

Dieser Befehl sucht nach der spezifizierten User ID in Ihrem Schlüsselbund und entfernt diesen Schlüssel, wenn ein passender gefunden wird. Denken Sie daran, daß jedes Fragment einer User ID zum Suchen verwendet werden kann. Die optionale Schlüsselbunddatei wird standardmäßig als „pubring.pgp“ angenommen. Geben Sie „secring.pgp“ an, wenn Sie Schlüssel aus Ihrem geheimen Schlüsselbund entfernen wollen. Sie können aber auch einen anderen Namen für die Schlüsselbunddatei angeben. Standardendung ist „.pgp“.

Wenn ein Schlüssel mehr als eine User ID besitzt, werden Sie gefragt, ob Sie eine von diesen entfernen wollen, aber den Schlüssel im Bund belassen wollen.

5.8.4 Einen Schlüssel aus dem Schlüsselbund kopieren

Um einen Schlüssel zu kopieren

```
pgp -kx userid Schluesseldatei [Schluesselbund]
```

Dieses Kommando kopiert den durch die User ID spezifizierten Schlüssel in eine separate Schlüsseldatei. Verwenden Sie diese Funktion, um Ihren öfffentlichen Schlüssel weiterzugeben.

Wenn der Schlüssel Unterschriften hat, werden diese mitkopiert.

Wenn Sie einen Schlüssel in druckbaren ASCII–Zeichen exportieren wollen, z.B. für E–Mail, geben Sie die Option `-kxa` an.

5.8.5 Den Schlüsselbund ansehen

Um den Inhalt Ihres Schlüsselbundes anzusehen:

```
pgp -kv[v] [userid] [Schluesselbund]
```

Alle Schlüssel des Bundes, die zur User ID passen werden aufgelistet. Wenn Sie User ID weglassen, wird der gesamte Inhalt des Schlüsselbundes angezeigt. Wenn Sie den optionalen Schlüsselbundnamen weglassen, wird „pubring.pgp“ benutzt. Sie können auch „secring.pgp“ angeben, um Ihre geheimen Schlüssel anzusehen oder einen anderen Schlüsseldateinamen. Standardextension ist „.pgp“.

Später werden wir das Konzept erklären, Schlüssel mit Unterschriften zu versehen. Um alle zu einem Schlüssel gehörigen Unterschriften anzusehen, benutzen Sie die Option `-kvv`:

```
pgp -kvv [userid] [Schluesselbund]
```

Es gibt auch noch eine andere Möglichkeit:

```
pgp Schluesseldatei
```

Ohne Zusatzkommandos wird PGP alle in der Datei enthaltenen Schlüssel anzeigen und fügt Sie in Ihren Schlüsselbund ein, wenn sie nicht schon vorhanden sind.

5.8.6 Wie schützen Sie öffentliche Schlüssel

In einem RSA–Kryptosystem müssen Sie die öffentlichen Schlüssel nicht vorm Bekanntwerden schützen. Es ist sogar besser, wenn sie so weit wie möglich verbreitet sind. Aber es ist sehr wichtig, öffentliche Schlüssel vor Veränderungen zu schützen, um sicher zu gehen, daß ein Schlüssel wirklich dem gehört, dem er zu gehören scheint. Dies ist der verletzlichste Punkt eines RSA–Kryptosystems. Zuerst sehen wir uns eine mögliche Katastrophe an, und dann wie man sie mit PGP sicher verhindert.

Nehmen wir an, Sie wollen eine private Nachricht an Alice schicken. Sie laden sich Alices öffentlichen Schlüssel aus einer Mailbox (oder BBS) herunter. Nachdem Sie die Nachricht mit diesem Schlüssel verschlüsselt haben, senden Sie diese über die E–Mail Station der Mailbox an Alice.

Unglücklicherweise, Ihnen und Alice unbekannt, hat ein anderer Benutzer namens Charlie die BBS infiltriert. Er hat ein neues Schlüsselpaar generiert und Alices User ID benutzt. Er vertauscht seinen gefälschten Schlüssel mit Alices echtem öffentlichem Schlüssel. Sie benutzen unwissentlich Charlies Fälschung anstatt Alices echten. Alles sieht normal aus, weil die Fälschung Alices User ID hat. Jetzt kann Charlie die für Alice bestimmte Nachricht entschlüsseln, da er den passenden geheimen Schlüssel hat. Er könnte sogar die Nachricht wieder mit Alices echtem öffentlichen Schlüssel verschlüsseln und an Alice senden, so daß niemand irgendetwas bemerkt. Er kann sogar augenscheinlich echte Unterschriften von Alice erzeugen, da ja jeder den gefälschten Schlüssel benutzt, um die Unterschriften zu prüfen.

Die einzige Möglichkeit, diese Katastrophe abzuwenden ist, jeden davon abzuhalten, die öffentlichen Schlüssel zu verändern. Wenn sie Alices Schlüssel direkt von Alice haben, ist das kein Problem. Aber das könnte schwierig sein, denn vielleicht lebt Alice tausend Kilometer entfernt oder ist zur Zeit nicht erreichbar.

Vielleicht können sie Alices Schlüssel von einem vertrauenswürdigen Freund, David, bekommen. David weiß, er hat den echten öffentlichen Schlüssel von Alice. David könnte Alices Schlüssel unterschreiben, um dessen Integrität zu bestätigen. David würde diese Unterschrift mit seinem eigenen geheimen Schlüssel erzeugen.

Das würde ein unterschriebenes öffentliches Schlüsselzertifikat erzeugen und zeigen, daß Alices Schlüssel nicht verändert wurde. Dazu müssen sie einen echten öffentlichen Schlüssel von David haben, um die Unterschrift zu prüfen. Vielleicht könnte David Alice ein beglaubigtes Zertifikat Ihres Schlüssels übergeben. David wirkt also als ein „Vorsteller“ zwischen Ihnen und Alice.

Der beglaubigte Schlüssel von Alice könnte von David oder Alice auf eine BBS geladen werden, von der Sie ihn später wieder herunterladen können. Sie können Davids Unterschrift mit seinem öffentlichen Schlüssel überprüfen. Niemand wird Sie dazu bringen, seinen gefälschten Schlüssel als Alices zu akzeptieren, weil niemand Davids Beglaubigung fälschen kann.

Eine vertrauenswürdige Person kann sich sogar darauf spezialisieren, neue Benutzer „bekanntzumachen“, indem er deren Schlüsselzertifikate beglaubigt. Diese Person könnte als „Schlüsselverwalter“ oder „Beglaubiger“ agieren. Von jedem von ihm beglaubigten Schlüsselzertifikat kann mit Sicherheit gesagt werden, daß es wirklich demjenigen gehört, dem es zu gehören scheint. Alle Benutzer, die an diesem Service teilnehmen wollen brauchen nur eine bekannt echte Kopie des öffentlichen Schlüssels vom „Schlüsselverwalter“, um dessen Unterschriften prüfen zu können.

Ein vertrauenswürdiger Schlüsselverwalter oder Beglaubiger ist besonders für große zentral kontrollierte Firmen oder Regierungsbehörden zu empfehlen. Einige Institutionen verwenden Hierarchien von Beglaubigern.

In kleineren Umgebungen würde es besser funktionieren, wenn man allen Nutzern erlaubt, als „Bekanntmacher“ für ihre Freunde zu fungieren. PGP geht diesen „organischen“ nicht–institutionellen Weg. Er spiegelt besser die Art wieder, wie Menschen sonst miteinander umgehen und erlaubt den Menschen die Wahl, wem sie das Schlüsselmanagement anvertrauen.

Die Notwendigkeit, öffentliche Schlüssel vor unerlaubter Veränderung zu schützen ist das schwierigste Problem in praktischen RSA–Applikationen. Es ist die Achillesferse der RSA–Kryptografie und ein großer Teil der Software–Komplexität wird nur gebraucht, um dieses eine Problem zu lösen.

Sie sollten einen öffentlichen Schlüssel nur dann benutzen, wenn Sie sichergestellt haben, daß er wirklich dem gehört, von dem er es behauptet. Sie können sich dessen sicher sein, wenn Sie ihn direkt vom Besitzer haben oder wenn er eine Unterschrift von jemandem trägt, von dem Sie schon wissen, daß Sie seinen echten öffentlichen Schlüssel haben. Auch sollte die User ID den vollen Namen des Besitzers enthalten, nicht nur den Vornamen.

Wie verlockt Sie auch immer sein mögen – und Sie werden verlockt sein – trauen Sie niemals, NIEMALS einem öffentlichen Schlüssel, den sie von einer BBS heruntergeladen haben, wenn er nicht von jemandem, dem sie trauen unterschrieben ist. Ein nicht beglaubigter Schlüssel kann von jedem verändert worden sein, sogar vom SysOp der Mailbox.

Wenn Sie gebeten werden, einen öffentlichen Schlüssel zu beglaubigen, stellen Sie sicher, daß er wirklich der Person gehört, die in der User ID angegeben ist, weil Ihre Unterschrift auf diesem Schlüssel Ihr Versprechen ist, daß dieser Schlüssel wirklich ihr gehört. Andere Leute werden diesem öffentlichen Schlüssel vertrauen, weil sie Ihrer Unterschrift darauf vertrauen. Es wäre falsch, Hörensagen zu glauben – unterschreiben Sie nur, wenn Sie aus erster Hand wissen, daß er wirklich ihr gehört. Vorzugsweise sollten Sie einen Schlüssel nur dann unterschreiben, wenn Sie ihn direkt von der Person haben.

Um einen Schlüssel zu unterschreiben, müssen Sie sich über den Eigentümer viel sicherer sein, als wenn Sie nur mit diesem Schlüssel verschlüsseln. Um sicher zu sein, daß ein Schlüssel zur Benutzung gut genug ist, sollten die Unterschriften einiger Bekanntmacher ausreichen. Aber um ihn selbst zu unterschreiben müssen Sie aus erster Hand wissen, wem er gehört. Vielleicht können Sie den Eigentümer anrufen und die Schlüsseldatei mit ihm überprüfen und festzustellen, daß Sie wirklich Ihren Schlüssel haben – und stellen Sie sicher, daß Sie mit der richtigen Person reden. Weiteres lesen Sie im Abschnitt „Überprüfen eines öffentlichen Schlüssels über Telefon" im Teil II.

Halten Sie sich vor Augen, daß Ihre Unterschrift auf dem öffentlichen Schlüssel nicht die Integrität dieser Person garantiert, sondern nur für die Integrität (das Eigentum) des Schlüssels bürgt. Sie werden nicht Ihre Reputation verlieren, weil Sie den Schlüssel eines Psychopaten beglaubigt haben, nur weil Sie absolut sicher waren, daß dieser Schlüssel wirklich ihm gehört. Andere Leute werden dem Schlüssel vertrauen, weil Sie Ihrer Unterschrift vertrauen (angenommen Sie vertrauen Ihnen), aber Sie müssen nicht dem Besitzer des Schlüssels vertrauen. Einem Schlüssel zu vertrauen bedeutet nicht, dem Schlüsselbesitzer zu vertrauen.

Vertrauen ist nicht unbedingt übertragbar; Ich habe einen Freund, von dem ich glaube, das er nicht lügt. Er ist ein leichtgläubiger Mensch, der glaubt, der Präsident lügt nicht. Das bedeutet nicht, daß ich glauben muß, der Präsident lügt nicht. Das ist ganz einfach. Wenn ich Alices Beglaubigung traue, und Alice traut Charlies Beglaubigung, heißt das nicht, daß ich Charlies Beglaubigung trauen muß.

Es wäre eine gute Idee, Ihren eigenen Schlüssel mit einer Sammlung von Unterschriften einiger Beglaubiger zu haben, in der Hoffnung, die meisten Leute werden wenigstens einem der Bekanntmacher genug trauen, für Ihren Schlüssel zu bürgen. Sie können Ihren beglaubigten Schlüssel in einigen Mailboxen ablegen. Wenn Sie jemandes Schlüssel beglaubigen, senden Sie ihm eine Kopie, damit er Ihre Unterschrift in seine Schlüsseldatei aufnehmen kann.

Achten Sie darauf, daß niemand Ihren eigenen öffentlichen Schlüsselbund verändern kann. Das Überprüfen neuer Schlüssel ist unbedingt von der Integrität der Schlüssel ihres Schlüsselbundes abhängig. Behalten Sie physikalische Kontrolle über Ihren geheimen Schlüsselbund, bevorzugterweise auf Ihrem eigenen PC, nicht auf einem entfernten (z.B. Netzwerserver), genauso wie Sie es mit Ihrem geheimen Schlüsselbund machen. Dies, um die Schlüssel vor Veränderung zu schützen, nicht um sie geheimzuhalten. Behalten Sie immer Sicherheitskopien ihrer Schlüsselbunde auf schreibgeschützten Medien.

Weil Ihr eigener öffentlicher Schlüssel als „oberste Autorität" zum direkten oder indirekten Beglaubigen aller anderen Schlüssel benutzt wird, ist er der wichtigste zu schützende Schlüssel. Um unerwünschte Veränderungen zu erkennen, können sie PGP anweisen, Ihren Schlüssel mit einer Sicherheitskopie auf einem schreibgeschützten Medium gegenzutesten. Eine genauere Beschreibung dieser Möglichkeit finden Sie in der Anleitung zum `-kc`–Kommando „Schlüsselbundüberprüfung" im Teil II der Dokumentation.

PGP nimmt generell an, daß sie physische Sicherheit über Ihr Sytem, Ihre Schlüsselbunde und natürlich PGP selbst haben. Wenn ein Angreifer PGP selbst verändern kann, kann er eventuell die Sicherheitsmaßnahmen ausschalten, die PGP hat, Ihre Schlüssel zu schützen.

Ein etwas komplizierter Weg, Ihren Schlüsselbund zu schützen wäre, wenn Sie die gesamte Schlüsseldatei mit Ihrem geheimen Schlüssel unterschreiben. Sie können eine separate Unterschriftsdatei mit dem Befehl `-sb` erzeugen (siehe Abschnitt: „Unterschriften von Nachrichten abtrennen" im Teil II). Unglücklicherweise brauchen Sie immer noch eine Sicherheitskopie Ihres öffentlichen Schlüssels, um die Integrität der Beglaubigung zu überprüfen. Benutzen Sie dazu nicht den Schlüssel aus der beglaubigten Unterschriftsdatei, denn diesen wollen Sie ja gerade überprüfen.

5.8.7 Wie erkennt PGP ob Schlüssel in Ordnung sind?

Bevor Sie diesen Abschnitt lesen, sollten Sie den Abschnitt „Wie schützen Sie Ihre öffentlichen Schlüssel" (siehe oben) lesen.

PGP stellt automatisch fest, welche Schlüssel in Ihrem öffentlichem Bund von Leuten beglaubigt sind, denen Sie trauen. Alle was Sie zu tun haben ist, PGP mitzuteilen, welchen Leuten Sie als Bekanntmacher trauen und deren Schlüssel mit Ihrem eigenen absolut vertrauenswürdigen Schlüssel zu beglaubigen. PGP kann so von diesen Leuten beglaubigte Schlüssel automatisch gültig machen. Und natürlich können Sie auch selbst Schlüssel beglaubigen. Mehr dazu später.

Es gibt zwei völlig unabhängige Kriterien, nach denen PGP entscheidet, ob ein öffentlicher Schlüssel in Ordnung ist – bringen Sie diese nicht durcheinander:

1. Gehört der Schlüssel wirklich dem, dem er zu gehören scheint? Mit anderen Worten, hat er eine vertraute Beglaubigung?
2. Gehört er jemandem, dem Sie trauen, wenn er andere Schlüssel beglaubigt?

PGP kann die Antwort auf die erste Frage selbst feststellen. Um die zweite Frage zu beantworten, braucht PGP Hilfe von Ihnen, dem Benutzer. Wenn Sie die zweite

Frage beantworten, kann PGP die Antwort auf die erste Frage selbst finden, wenn es feststellt, daß ein Benutzer einen neuen Schlüssel beglaubigt hat, dem Sie vertrauen.

Schlüssel, die von vertrauten Bekanntmachern beglaubigt sind, werden von PGP als gültig behandelt. Schlüssel, die vertrauten Bekanntmachern gehören, müssen selbst entweder von Ihnen oder anderen vertrauten Bekanntmachern beglaubigt sein.

PGP erlaubt Ihnen auch, verschiedene „Tauglichkeitsgrade“ für Bekannmacher zu vergeben. Ihr Vertrauen in Schlüsselbesitzer reflektiert nicht nur Ihre Meinung von deren persönlichen Integrität, sondern sollte auch darauf basieren, wie gut diese Leute die Schlüsselverwaltung beherrschen. Sie können eine Person für PGP deklarieren als -unbekannt, -nicht vertrauenswürdig, -vertrauenswürdig und -absolut vertrauenswürdig, andere Schlüssel zu beglaubigen. Diese „Vertrauensinformation“ wird in Ihrem Schlüsselbund auf den Schlüsseln gespeichert, wird aber von PGP nicht mitkopiert, wenn Sie Schlüssel extrahieren, weil Ihre persönliche Meinung als vertraulich angesehen wird.

Wenn PGP die Gültigkeit eines Schlüssel feststellt, überprüft es die „Vertrauenslevel“ der Beglaubigungen auf dem Schlüssel. Es errechnet einen gewichteten Vertrauensindex für die Gültigkeit, z.B. sind zwei vertrauenswürdige Beglaubigungen so gut wie eine absolut vertrauenswürdige. PGPs Skepsis ist einstellbar, Sie können es anweisen, daß zwei absolute oder drei vertrauenswürdige Beglaubigungen nötig sind, um einen Schlüssel gültig zu machen.

Ihr eigener Schlüssel ist für PGP „definitiv“ gültig, er benötigt keine Beglaubigungen um ihn absolut vertrauenswürdig zu machen. PGP weiß, welche öffentlichen Schlüssel Ihnen gehören, indem es die zugehörigen geheimen im Schlüsselbund sucht. PGP nimmt auch an, daß Sie sich selbst absolut vertrauen, andere Schlüssel zu beglaubigen.

Mit der Zeit werden Sie Schlüssel anderer Leute sammeln, die Sie als vertraute Bekanntmacher deklarieren. Jeder andere wird auch seine vertrauten Bekanntmacher wählen. Und jeder wird mit seinen Schlüssel eine Anzahl Beglaubigungen sammeln, in der Hoffnung, daß jeder, der ihn erhält wenigstens einer oder zwei vertraut. Dies wird ein dezentralisiertes fehlertolerantes Netz der Sicherheit für alle öffentlichen Schlüssel schaffen.

Diese einzigartige Vorgehensweise steht im scharfen Kontrast zu den Standardschemata, welche die Regierung benutzt, wie das „Internet Privacy Enhanced Mail“ (PEM) welche auf zentraler Kontrolle basieren. Diese Standardschemata verlassen sich auf eine Hierarchie von Zertifizierungsautoritäten, die vorschreiben, wem man zu vertrauen hat. PGPs dezentralisierte Wahrscheinlichkeitsmethode für die Feststellung der Gültigkeit von öffentlichen Schlüsseln ist das Herzstück seiner Schlüsselverwaltung. PGP läßt Sie auswählen, wem Sie vertrauen und setzt Sie auf die Spitze der Vertrauenspyramide. PGP ist für Menschen, die es bevorzugen, ihre Fallschirme selbst zu packen.

5.8.8 Wie schützen Sie geheime Schlüssel

Schützen Sie Ihren geheimen Schlüssel und seinen Sicherheitssatz gut. Wirklich, wirklich gut. Wenn Ihr geheimer Schlüssel je bekannt wird, teilen Sie das besser jeder interessierten Partei mit (viel Glück), bevor irgendjemand ihn benutzt, um in Ihrem Namen Unterschriften zu machen. Er könnte zum Beispiel benutzt werden, um gefälschte Schlüsselzertifikate zu beglaubigen, was vielen Leuten Probleme

bereiten könnte, speziell wenn Sie großes Vertrauen genießen. Und natürlich, ein Bekanntwerden Ihres geheimen Schlüssels gefährdet auch alle Nachrichten, die an Sie geschickt werden.

Um Ihren geheimen Schlüssel zu schützen, fangen Sie am besten damit an, immer physische Kontrolle über ihn zu behalten. Es ist OK, ihn auf Ihrem Heimcomputer zu speichern, oder Sie können ihn auf ihrem Notebook halten, das Sie mit sich herumtragen können. Wenn Sie einen Bürocomputer benutzen müssen, über den Sie nicht immer die Kontrolle haben, behalten Sie den Schlüssel auf einer schreibgeschützten Diskette, die Sie nicht im Büro zurücklassen sollten, wenn Sie gehen. Es ist keine gute Idee, Ihren Schlüssel auf einem Fern–PC, wie einem remote dial–in Unix–System zu haben. Jemand könnte Ihre Modem–Leitung abhören, Ihren Sicherheitssatz herausfinden und dann den Schlüssel vom Fern–PC holen. Benutzen Sie Ihren Schlüssel nur auf einem PC, den Sie kontrollieren können.

Speichern Sie Ihren Sicherheitssatz nicht auf dem gleichen Computer wie Ihre geheime Schlüsseldatei. Das wäre ungefähr so gefährlich, wie Ihre Geheimzahl in der Geldbörse mit Ihrer Scheckkarte aufzubewahren. Niemand anders sollte die Diskette mit der geheimen Schlüsseldatei in die Hände bekommen. Am besten wäre es, Sie merken sich den Sicherheitssatz gut und schreiben ihn nirgends auf. Wenn Sie ihn sich aufschreiben müssen, schützen Sie ihn gut, vielleicht sogar besser als Ihre geheime Schlüsseldatei.

Behalten Sie Sicherungsdateien Ihrer geheimen Schlüsseldatei – denken Sie daran, daß nur Sie die geheimen Schlüssel besitzen. Wenn Sie diese verlieren, werden alle von Ihnen verteilten öffentlichen Schlüssel nutzlos.

Das dezentralisierte Schlüsselmanagement hat seine Vorteile, aber das bedeutet auch, daß wir uns nicht auf eine zentrale Liste ungültig gemachter Schlüssel verlassen können. Das macht es ein wenig schwieriger, den Schaden gering zu halten, wenn der geheime Schlüssel bekannt wird. Sie müssen einfach in den Wald rufen und hoffen, daß jeder es hört.

Wenn der schlimmste Fall eintritt – Ihr geheimer Schlüssel und Ihr Sicherheitssatz fallen in falsche Hände (hoffentlich finden Sie das irgendwie heraus) – müssen Sie ein „Schlüssel bekannt geworden“–Zertifikat ausgeben. Dieses Zertifikat warnt andere Leute vor der Benutzung Ihres öffentlichen Schlüssels. PGP kann mit dem Befehl `-kd` ein solches Zertifikat für Sie erstellen. Sie müssen es dann irgendwie zu jedem auf diesem Planeten schicken oder wenigstens zu Ihren Freunden und deren Freunden, usw. Deren PGP wird dieses Zertifikat in die öffentlichen Schlüsselringe einbauen und sie automatisch davon abhalten, ihren öffentlichen Schlüssel zu benutzen. Sie können jetzt ein neues Schlüsselpaar erzeugen und den neuen öffentlichen Schlüssel mit dem Ungültigkeitszertifikat absenden.

6 Erweiterte Themen

Die meisten erweiterten Funktionen von PGP werden im „PGP User's Guide, Teil II“ behandelt. Aber es gibt noch einige, die hier erwähnt werden sollen.

6.1 Verschlüsselte Texte über E–Mail: Das Radix–64–Format

Viel E–Mail–Systeme lassen nur Texte im ASCII–Format zu, nicht die 8–bit Binärdaten, aus denen die Chiffrate bestehen. Um dieses Problem zu umgehen, unterstützt PGP das ASCII Radix–64–Format für Chiffratdateien, das ähnlich dem „Internet Privacy–Enhanced Mail" (PEM) oder dem Internet MIME–Format ist. Dieses spezielle Format repräsentiert 8–bit Binärdaten mit druckbaren ASCII–Zeichen und ist so nützlich, um verschlüsselte binäre Daten über 7–bit Kanäle oder normale E–Mail zu versenden. Dieses Format funktioniert als eine Art „Transportpanzer", der die Daten beim Durchgang durch das Internet vor Veränderung schützt. PGP benutzt auch eine CRC, um Übertragungsfehler herauszufiltern.

Radix–64 konvertiert die Daten, indem Gruppen von jeweils 3 Binärbytes in 4 druckbare Zeichen umgewandelt werden, also werden die Daten um ca. 33% größer. Aber diese Größenzunahme ist nicht so schlimm, wenn Sie bedenken, das PGP die Datei meist vorher um mehr als das komprimiert hat, bevor sie verschlüsselt wurde.

Um ein Chiffrat im Radix–64–Format zu erzeugen, hängen Sie die Option `a` an, wenn Sie eine Nachricht verschlüsseln oder unterschreiben, z.B.:

```
pgp -esa nachricht.txt seine_userid
```

Dieses Beispiel erzeugt ein Chiffrat namens „nachricht.asc", das die Daten im MIME–ähnlichen Radix–64 Format enthält. Diese Datei kann leicht in einen Texteditor oder 7–Bit–Kanäle geladen oder als normale E–Mail verschickt werden.

Eine Radix–64 transportgeschützte Nachricht zu entschlüsseln ist kein Unterschied zur normalen Vorgehensweise. z.B.:

```
pgp nachricht
```

PGP sucht automatisch erst nach der ASCII–Datei 'nachricht.asc', bevor es nach der Binärdatei „nachricht.pgp" sucht. Es erkennt, daß die Datei Radix–64 ist und konvertiert sie nach binär zurück, wobei es als Nebenprodukt eine „.pgp" in Binärform erzeugt. Die endgültige Ausgabe ist der normale Text, wie er in der Datei „nachricht.txt" war.

Die meisten E–Mail Stationen verbieten das Senden von Nachrichten, die länger als 50000 oder 65000 Bytes sind. Längere Nachrichten müssen in kleinere Blöcke gespalten und einzeln übertragen werden. Wenn die verschlüsselte Nachricht sehr lang ist, und sie Radix–64 verlangen, spaltet PGP sie automatisch in Blöcke auf, die klein genug sind, um sie per E–Mail zu übertragen. Diese Blöcke werden in Dateien mit den Endungen „.as1', „.as2", „.as3", usw. aufgeteilt. Der Empfänger muß diese einzelnen Dateien wieder in der richtigen Reihenfolge zu einer großen zusammenfügen, bevor er sie entschlüsseln kann. Während der Entschlüsselung ignoriert PGP alle Texte, die nicht in den Radix–64–Blocks eingeschlossen sind.

Wenn Sie einen öfffentlichen Schlüssel im Radix-64 verschicken wollen, hängen Sie einfach die Option `-a` an, wenn Sie den Schlüssel aus dem Bund herauskopieren.

Wenn Sie vergessen haben, die `-a`–Option zu benutzen, können Sie das Chiffrat immer noch direkt ins Radix–64 konvertieren, indem Sie die Option `-a` allein benutzen, ohne eine Verschlüsselungsoption anzugeben. PGP gibt der konvertierten Datei die Endung „.asc".

Wenn Sie eine Nachricht unterschreiben, ohne sie zu verschlüsseln, wird PGP die Datei normalerweise komprimieren, wodurch sie aber für den menschlichen Leser unkenntlich wird. Das wäre nicht weiter störend, wenn die unterschriebene Datei nur archiviert werden soll. Aber wenn Sie die unterschriebene Nachricht als E–Mail verschicken wollen, und die Nachricht ist in Text–(nicht Binär)form, gibt es eine Möglichkeit, die Datei unkomprimiert zu versenden und das Radix–64 nur auf die binäre Unterschrift anzuwenden, nicht auf die Nachricht selbst. Dadurch kann der Empfänger die Nachricht auch ohne Hilfe von PGP lesen. Natürlich braucht er trotzdem PGP, wenn er die Unterschrift prüfen will. Mehr Informationen zu diesem Thema finden Sie in der Erklärung zum Parameter CLEARSIG im Abschnitt „Konfigurationsparameter setzen" im Teil II der Dokumentation.

Manchmal wollen Sie vielleicht einfach nur binäre Daten über E–Mail versenden, die nicht mit PGP verschlüsselt oder unterschrieben sind. Manche Leute benutzen das Unix–Utility uuencode dafür. PGP kann auch dafür benutzt werden, wenn Sie nur die Option `-ar` verwenden, außerdem arbeitet es besser als uuencode. Weitere Details finden sie im Abschnitt „PGP als besseres Uuencode verwenden" im Teil II.

7 Angreifbarkeit

Kein Datensicherheitssystem ist unangreifbar. PGP kann auf verschiedenen Wegen hintergangen werden. Mögliche Angreifbarkeiten schließen ein: Bekanntwerden Ihres geheimen Schlüssels oder des Schutzsatzes, Veränderung öffentlicher Schlüssel, gelöschte Dateien, die immer noch auf Ihrer Festplatte sind, Viren und Trojanische Pferde, Lücken in der physikalischen Sicherheit, elektromagnetische Emissionen, ungeschützte Multi–User–Systeme, Datenverkehranalyse oder sogar direkte Kryptoanalyse.

Eine detaillierte Erörterung zu diesem Thema finden Sie im Abschnitt „Angreifbarkeiten" im Teil II.

Literatur

[1] c't – Krypto–Kampagne. http://www.heise.de/ct/pgpCA.

[2] Internationale PGP–Homepage. http://www.pgpinternational.com.

[3] Deutsche PGP–Dokumentation. ftp://ftp.pgp.net/pub/pgp/doc/german.

A Schnellreferenz

In diesem Abschnitt finden Sie eine Zusammenfassung der PGP–Befehle.

Um eine Nachricht mit dem öfentlichen Schlüssel des Empfängers zu verschlüsseln:

```
pgp -e textdatei seine_userid
```

Um eine Nachricht mit Ihrem geheimen Schlüssel zu unterschreiben:

```
pgp -s textdatei [-u ihre_userid]
```

Um einen ASCII–Text zu unterschreiben und eine über E–Mail versendbare lesbare Nachricht zu erzeugen:

```
pgp -sta textdatei [-u ihre_userid]
```

Um eine Nachricht mit Ihrem geheimen Schlüssel zu unterschreiben und sie dann mit dem öffentlichen Schlüssel des Empfängers zu verschlüsseln:

```
pgp -es textdatei seine_userid [-u ihre_userid]
```

Um eine Nachricht konventionell zu verschlüsseln:

```
pgp -c textdatei
```

Um eine verschlüsselte Datei zu entschlüsseln oder eine Beglaubigung zu überprüfen:

```
pgp chiffratdatei [-o ausgabedatei]
```

A.1 Kommandos zur Schlüsselverwaltung

Um Ihr eigenes individuelles Schlüsselpaar zu erzeugen:

```
pgp -kg
```

Um den Inhalt einer Schlüsseldatei zu Ihrem Schlüsselbund hinzuzufügen:

```
pgp -ka schluesseldatei [schluesselbund]
```

Um einen Schlüssel aus dem geheimen oder öffentlichen Schlüsselbund herauszukopieren:

```
pgp -kx userid schluesseldatei [schluesselbund]
```

oder:

```
pgp -kxa userid schluesseldatei [schluesselbund]
```


Um den Inhalt eines Schlüsselbundes anzusehen:

```
pgp -kv[v] [userid] [schlsselbund]
```

Um einen „Fingerabdruck“ eines öffentlichen Schlüssels anzusehen, um ihn mit seinem Besitzer über Telefon zu überprüfen.

```
pgp -kvc [userid] [schluesselbund]
```

Um den Inhalt des öffentlichen Schlüsselbundes anzusehen, und die Beglaubigungen der Schlüssel zu überprüfen:

```
pgp -kc [userid] [schluesselbund]
```

Um die UserID oder den Schutzsatz für Ihren geheimen Schlüssel zu edieren:

```
pgp -ke userid [schluesselbund]
```

Um die Vertrauensparameter eines öffentlichen Schlüssel zu edieren:

```
pgp -ke userid [schluesselbund]
```

Um einen Schlüssel oder eine UserID aus Ihrem Schlüsselbund zu entfernen:

```
pgp -kr userid [schluesselbund]
```

Um einen Schlüssel in Ihrem öffentlichen Schlüsselbund zu beglaubigen:

```
pgp -ks seine_userid [-u ihre_userid] [schluesselbund]
```

A.2 Optionen in Verbindung mit anderen Kommandos

Um ein Chiffrat im Radix–64 ASCII–Format zu erzeugen, fügen Sie die Option `-a` an:

```
pgp -sea textdatei seine_userid
```

oder:

```
pgp -kxa userid schluesseldatei [schluesselbund]
```

Um einen Unix–artigen Filtermodus zu verwenden, die Eingaben von der Standardeingabe zu lesen und die Ausgaben zur Standardausgabe zu schicken, fügen Sie die Option `-f` an:

```
pgp -feast seine_userid <eingabedatei >ausgabedatei
```