LaTeX-Vorlage für diverse Ausarbeitungen – oder so ähnlich

Projekt-/Studien-/Bachleorarbeit

für die Prüfung zum

Bachelor of Science

des Studienganges Informatik / Informationstechnik

an der

Dualen Hochschule Baden-Württemberg Karlsruhe

von

Max Mustermann

Abgabedatum 1. April 2090

Bearbeitungszeitraum	12 Wochen
Matrikelnummer	4711
Kurs	tinf17b3
Ausbildungsfirma	Firmenname Stadt
Betreuer der Ausbildungsfirma	Titel Vorname Nachname
Gutachter der Studienakademie	Titel Vorname Nachname

Erklärung

Ich versichere hiermit, dass ich meine Projekt-/Studien-/Bachleorarbeitmit dem Thema: »LaTeX-Vorlage für diverse Ausarbeitungen
–
oder so ähnlich« selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe. Ich versichere zudem, dass die eingereichte elektronische Fassung mit der gedruckten Fassung übereinstimmt. _____

_____

Ort Datum Unterschrift

Sofern vom Dualen Partner ein Sperrvermerk gewünscht wird, ist folgende Formulierung zu verwenden:

Sperrvermerk

Der Inhalt dieser Arbeit darf weder als Ganzes noch in Auszügen Personen außerhalb des Prüfungsprozesses und des Evaluationsverfahrens zugänglich gemacht werden, sofern keine anderslautende Genehmigung vom Dualen Partner vorliegt.

Zusammenfassung

Dieses LaTeX-Dokument kann als Vorlage für einen Praxis- oder Projektbericht, eine Studien- oder Bachelorarbeit dienen.

Zusammengestellt von Prof. Dr. Jürgen Vollmer <juergen.vollmer@dhbw-karlsruhe.de>
`https://www.karlsruhe.dhbw.de`. Die jeweils aktuellste Version dieses LaTeX-Paketes ist immer auf der *FAQ-Seite* des Studiengangs Informatik zu finden: `https://www.karlsruhe.dhbw.de/inf/studienverlauf-organisatorisches.html` → *Formulare und Vorlagen*.

Stand `$Date: 2020/03/13 15:07:45 $`

Inhaltsverzeichnis

Abbildungsverzeichnis

Tabellenverzeichnis

Liste der Algorithmen

Formelverzeichnis

Abkürzungsverzeichnis

Kapitel 1

Grundlagen

In diesem Kapitel werden die theoretischen Grundlagen der Kryptologie, Mathematik, Komplexitätstheorie und der IT-Sicherheit erläutert, die in dieser Arbeit eine Rolle spielen. Aus diesen sollen die grundlegenden Funktionen eines kleptographischen Angriffes und dessen Folgen abgeleitet werden.

1.1 Kryptologie

Die Kryptologie ist die wissenschaftliche Disziplin für den Schutz von Daten. Unter ihr stehen die zwei Felder der Kryptographie und der Kryptoanalyse.

1.1.1 Kryptographie

Die Kryptologie befasst sich mit der Entwicklung von Verfahren und Techniken für den sicheren Austausch von Daten. Dabei stehen zwei Eigenschaften im Fokus:

Eigenschaften

Geheimhaltung Durch Geheimhaltung sollen, bei der Übertragung von Daten zwischen Teilnehmern, Unbeteiligte keine Erkenntnisse über den Inhalt erlangen. Dies kann durch physikalische oder organisatorische Maßnahmen erreicht werden, wobei Unbeteiligten der Zugang zu den übertragenen Daten verwehrt wird. Diese Maßnahmen sind sinnvoll bei der Übergabe der Daten in einer nicht digitalen Welt. Bei der Kommunikation in digitalen Netzen, wie u.a. dem Internet, sind diese Maßnahmen nur schwer zu implementieren. Dies gilt nicht für kryptographische Maßnahmen. Dabei ist es nicht mehr das Ziel, Unbeteiligten den Zugang zu den übertragenen Daten zu erschweren, sondern den Inhalt der Daten während der Übertragung zu verschlüsseln. Dadurch soll es Unbeteiligten nahezu unmöglich sein, aus den mitgehörten oder abgefangenen Daten, Rückschlüsse auf deren Inhalt zu erlangen.[1]

Authentifikation Durch Authentifikation soll es den Teilnehmern einer Kommunikation möglich sein, die anderen Teilnehmer und empfangene Nachrichten zweifelsfrei identifizieren und zuweisen zu können. Hierbei spielen Signaturverfahren eine wichtige Rolle, da kein Geheimnis

[1] Beutelspacher, Schwenk und Wolfenstetter 2015, S. 1.

benötigt wird um einen Teilnehmer zu authentifizieren. Dabei können sich Teilnehmer durch das Wissen oder den Besitz eines Geheimnisses (Passwort, Zertifikat, Schlüssel) authentifizieren.[2]

Nur wenn beide Eigenschaften gegeben sind ist eine Übertragung von Daten als sicher anzusehen. Falls die Geheimhaltung fehlt, kann der Inhalt durch Sniffing mitgelesen werden. Falls die Authentifikation der Teilnehmer fehlt, können sich Unbeteiligte als ëchte"Teilnehmer ausgeben und somit die Daten an ihrem Endpunkt entschlüsseln.

Zusätzliche Eigenschaften

Perfect Forward Security Perfect Forward Security

Kryptographische Verfahren

Kryptographische Verfahren sind Algorithmen, welche die Genheimhaltung von Daten und die Authentifikation von Teilnehmers und Nachrichten sicherstellt. Dadurch kann man sie in Verschlüsselungsverfahren und Authentifikationsverfahren unterscheiden. Dabei können Verfahren, wie z.B. Rivest-Shamir-Adleman (RSA) beiden Aufgaben übernehmen.

Asymmetrische Verschlüsselung Bei asymmetrischen Verschlüsselungsverfahren wird statt dem gleichen Schlüssel für das Ver- und Entschlüsseln, zwei verschiedene Schlüssel verwendet. Dabei hat jeder Teilnehmer ein öffentlichen Schlüssel e und einen privaten und geheimen Schlüssel d. Hierbei ist es vorgesehen, dass möglichst alle potenziellen Teilnehmer den Schlüssel e kennen. Wenn eine Nachricht mit einem der beiden Schlüssel chiffriert wurde, kann nur mittels dem anderen Schlüssel dechiffriert werden. Somit können Nachrichten an einen Teilnehmer verschlüsselt versandt werden, indem diese mit dem öffentlichen Schlüssel e des Teilnehmers chiffriert wird. Nun kann nur der Teilnehmer mit dem zugehörigen privaten Schlüssel d, die Nachricht verschlüsseln. Zusätzlich kann auch die Authentifikation von Teilnehmer und die Authentizität von Nachrichten mit hoher Sicherheit festgestellt werden. Somit kann der Author einer Nachricht, einen Fingerabdruck dieser Nachricht mit seinem privaten Schlüssel d signieren, an die Nachricht anhängen und dann beide Teile verschlüsseln. Diese Signatur kann verifiziert werden, indem der Empfänger die Nachricht entschlüsselt und dann die Signatur verifiziert, indem er den öffentlichen Schlüssel des Authors e auf diesen anwendet. Danach vergleicht er den empfangenen Fingerabdruck mit einem eigens erstellten Fingerabdruck. Somit kann die Geheimhaltung, Authentizität und Integrität der Nachricht bestimmt werden.

Die zwei Schlüssel e und d eines Teilnehmers, werden auch als Schlüsselpaar bezeichnet. Ein solches verfahren, wird asymmetrisch genannt, da für das Ent- und Verschlüsseln zwei unterschiedliche Informationen vorliegen müssen. Diese Informationen sind auch nicht auseinander ableitbar, wie es z.B. bei multiplikativen Chiffren der Fall wäre. Die Funktionalität des Verfahrens, beruht auf der Annahme, dass alle Teilnehmer Zugang zu den öffentlichen Schlüssel jedes anderen Teilnehmers haben bzw. haben können. Durch diese Charakteristika werden solche Verfahren auch als Public-Key-Kryptographie bezeichnet.

Dabei wird stets die Annahme getroffen, dass der private Schlüssel eines Teilnehmers ausschließlich diesem vorliegt. Anderenfalls ist die Geheimhaltung und die Authentifikation beim Informationsaustausch von und mit diesem Teilnehmer nicht mehr gewährleistet. Somit wäre die Sicherheit kompromittiert.

Die Schlüssel eines Schlüsselpaar bilden somit Umkehrfunktionen zueinander.

[2] Beutelspacher, Schwenk und Wolfenstetter 2015, S. 2.

Hybride Verfahren Asymmetrische Verschlüsselungsverfahren haben häufig den Nachteil, dass die deutlich rechenaufwändiges sind, wie wir später bei RSA sehen werden. Es liegen zwar effiziente Verfahren vor, um z.B. die modularen Potenz aus zwei 300-stelligen Zahlen und einem Modulo zu bilden 1.2.3. Dennoch sind diese Verfahren mit mehr Aufwand verbunden, als z.B. symmetrische Blockchiffren wie Advanced Encryption Standard (AES).

Deshalb werden asymmetrische Verfahren für die Initialisierung der Kommunikation verwendet. In dieser Initialisierungsphase soll der Teilnehmer authentifiziert werden und ein gemeinsamer, geheimer, symmetrischer Schlüssel vereinbart werden.

In der darauffolgenden Kommunikationsphase werden die Nachrichten durch symmetrische Chiffren mittels des vereinbarten Schlüssels, effizient verschlüsselt und auf der Gegenseite entschlüsselt. Asymmetrisch Chiffren werden hier benutzt um Fingerabdrücke von Nachrichten zu signieren und zu verifizieren, wie oben 1.1.1 gezeigt. Zusätzlich werden durch asymmetrische Verfahren regelmäßig neue symmetrische Schlüssel vereinbart.

Solche hybriden Verfahren sind z.B. beim Browsen im Internet zu finden. Hier ein Beispiel: $TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256$

Angriffe

Known Cipher Attack

Known Plaintext Attack

Chosen Plaintext Attack

Chosen Cipher Attack

1.2 Mathematik

Mathematische Probleme stellen die Grundlage für moderne Kryptographie.

1.2.1 Diskreter Logarithmus

Bei der Bestimmung des Logarithmus wird der Exponent (hier: x) gesucht, welcher mit einer bekannten Zahl als Basis z, eine weitere bekannte Zahl y ergibt.

$$z^x = y \tag{1.1}$$

Der diskrete Logarithmus bezieht hier auf die Berechnung des Logarithmus in ein Gruppe. Diese Gruppe bildet sich aus der Rechnung mit Restklassen (modulo). Dadurch entsteht folgendes Problem, bei der die Variable x gesucht ist und alle anderen Variablen bekannt sind.

$$z^x \pmod n \equiv y \tag{1.2}$$

Hierbei ist in der Notation zu beachten, dass sich durch das Rechnen auf mit einer Gruppe, Äquivalenzklassen ($\equiv$) bilden. Diese entsprechen den Restklassen des Rechnen mit Modulo. n ist die Mächtigkeit der Aquivalenzklassen.

Die Bestimmung von x in 1.2 wird als Problem des diskreten Logarithmus bezeichnet. Mit der Komplexität wird sich in den Grundlagen der Komplexitätstheorie beschäftigt.

Dabei ist die Umkehrfunktion, des diskreten Logarithmus $f(x)$ 1.2, mathematisch einfach zu berechnen. Diese Umkehrfunktion entspricht der diskreten Exponentialfunktion:

$$f^{-1}(x) = z^x \pmod{n} \equiv y \tag{1.3}$$

Hierbei sind z, x, n gegeben und y gesucht.

1.2.2 Faktorisierung

Bei der Faktorisierung wird versucht eine Zahl in Faktoren zu zerlegen. Dabei handelt es sich, im Kontext der Kryptographie, meist um die Faktorisierung des Produkts zweier großer Primzahlen. Dadurch bildet sich folgende Formel, wobei p und q Primzahlen sind (also Element der Menge der Primzahlen $\mathbb{P}$) und n das resultierende Produkt:

$$n = p * q \mid p, q \in \mathbb{P} \tag{1.4}$$

Da n das Produkt zweier Primzahlen ist, sind seine einzigen Teiler: n selbst, 1 und die seine Primfaktoren p und q. Deshalb handelt es sich hierbei auch um eine Primfaktorzerlegung von n.

Dabei ist die Primfaktorzerlegung von n ein rechenaufwändiges Problem, falls p udn q große Zahlen sind. Im Gegensatz dazu ist die Berechnung von bzw. die Validierung mit n sehr einfach, da hierfür nur die Multiplikation von p und q notwendig ist. Somit liegt die gleiche Situation, wie beim Problem des diskreten Logarithmus 1.2.1 vor: Ein rechenaufwändiges Problem, dessen Umkehrfunktion sehr einfach ist[3].

1.2.3 Effiziente Berechnung der diskreten Exponentialfunktion

In der Kryptographie werden große Zahlen genutzt, um die Sicherheit der verwendeten Algorithmen zu gewährleisten. Hierfür wird als Beispiel angenommen, dass als Basis z eine 256-bit Lange Zahl hoch einem 300-bit langem Exponenten x genommen werden soll. Hierbei ist n 1024-bit lang.

Wenn mann nun z in Byte berechnet wäre dies eine 32 Byte lange Zahl.

x entspricht einer ungefähr 90. stelligen Zahl.

$$z^{10^{90}} \pmod{n} \equiv y \tag{1.5}$$

Eine numerische Berechnung von $z^{10^{90}}$ ist aufgrund von begrenzten Ressourcen nicht möglich.

Jedoch kann man sich die diskrete Eigenschaft dieser Problems sich zu nutzte machen. Hierfür können Verfahren, wie Square-and-Multiply zusammen mit der Restklassenberechnung genutzt werden. Dadurch lassen sich auch großzahlige Exponenten berechnen. Hierfür soll ein einfaches Beispiel gegeben werden:

$$37^{52} \pmod{128} \equiv y \tag{1.6}$$

Bei Betrachtung der Äquivalenzgleichung fällt auf, dass 37^{52} eine große Zahl ergibt. Jedoch wird diese Zahl noch $x \pmod 1)28$ gerechnet. Dadurch liegt das Ergebnis in einem Zahlenraum von:

$$y \in \mathbb{N} \mid 0 \leq y < 128 \tag{1.7}$$

[3] Beutelspacher, Schwenk und Wolfenstetter 2015, S. 179.

Auf Grundlage der Potenzgesetze wird 37^{52} nun zerlegt.

$$\begin{aligned} 52 &= 32 + 16 + 4 = 2^5 + 2^4 + 2^2 \\ 37^{52} \pmod{128} &\equiv 37^{2^5} * 37^{2^4} * 37^{2^2} \\ \equiv 37^{2^5} \pmod{128} * 37^{2^4} \pmod{128} &* 37^{2^2} \pmod{128} \end{aligned} \tag{1.8}$$

Die einzelnen Bestandteile werden dann iterativ berechnet und durch Multiplikation zusammengefasst (siehe 1.8). Dies wird als Square-and-Multiply-Verfahren bezeichnet.

$$\begin{aligned} 37^{2^2} \pmod{128} &\equiv (37^{2^1} \pmod{128})^2 \\ 37^{2^3} \pmod{128} &\equiv (37^{2^2} \pmod{128})^2 \\ 37^{2^4} \pmod{128} &\equiv (37^{2^3} \pmod{128})^2 \\ 37^{2^5} \pmod{128} &\equiv (37^{2^4} \pmod{128})^2 \end{aligned} \tag{1.9}$$

Allgemein

Gegeben mit gesucht y:

$$z^x \pmod{n} \equiv y \tag{1.10}$$

Zerlegung von x eine Summe von Zweierpotenzen:

$$x = 2^0 + 2^1 + 2^2 + ... \tag{1.11}$$

Dabei bilden die binären Logarithmen der einzelnen Zweierpotenzen die Menge $\mathbb{K}$.

Berechnung der einzelnen Faktoren durch iteratives Square-and-Multiply-Verfahren. Dies wird bis $f(max(\mathbb{K}))$ berechnet. $max(\mathbb{K})$ steht hier für das Element von $\mathbb{K}$, mit dem größten Wert.

$$f(i+1) = f(i)^2 \pmod{n} \mid f(1) = z^1 \pmod{n} \tag{1.12}$$

Zuletzt wird das Produkt, aller Ergebnisse von $f(x)$ für die Elemente der Menge $\mathbb{K}$, gebildet. Dabei gilt:

$$\prod_{k \in \mathbb{K}} f(k) \equiv z^x \pmod{n} \equiv y \tag{1.13}$$

1.3 Komplexitätstheorie

Die Komplexitätstheorie befasst sich mit der Komplexität von Problemen, welche durch Algorithmen gelöst werden. Dabei wird der Speicherbedarf und der Zeitaufwand eines Algorithmus. Schrankenfunktionen werden gebildet durch die Betrachtung des Speicherbedarf und des Zeitaufwands im Bezug auf die Länge der Eingabeparameter. Da hier eine reine kryptographische Betrachtung der Schrankenfunktionen stattfinden soll, wird hier nur in zwei verallgemeinerte Schrankenfunktionen[4] unterschieden:

- Polynomiale Komplexität
- Nichtpolynomiale Komplexität

[4] Beutelspacher, Schwenk und Wolfenstetter 2015, S. 178.

Polynomiale Komplexität umfasst hier alle Probleme, die algorithmisch mit polynomialem Aufwand (Zeit/Speicher) gelöst werden können. D.h. bei steigender Eingabelänge n steigt der Aufwand im schlimmsten Fall mit $\mathcal{O}(n^c \mid \text{c is constant})$. Diese Probleme gehören damit zur Komplexitätsklasse **P**. Diese umfasst alle Probleme, welche algorithmisch mit maximal polynomialem Aufwand gelöst werden können. Diese Probleme können meistens von modernen Computern gelöst werden.

Nichtpolynomiale Komplexität hingegen umfasst alle Probleme, die mehr als polynomialen Aufwand im Worst-Case brauchen. Dies können Probleme sein, die algorithmisch nur mit exponentiellen $\mathcal{O}(d^n \mid d > 1)$ oder faktoriellen $\mathcal{O}(n!)$ Aufwand[5] gelöst werden können. Diese Probleme werden der Komplexitätsklasse **NP** zugewiesen. Dies sind Probleme können nicht von deterministischen Computern in mit polynomialen Aufwand gelöst werden.

Bezug zur Kryptographie Dadurch sind sie für die Kryptographie besonders interessant, da man die Sicherheit eines Systems auf ein NP-vollständiges Problem stützen kann. Somit ist die theoretische Sicherheit des Systems nicht brechbar. Jedoch sollte darauf geachtet werden, dass die vorgesehenen Teilnehmer an einem Datenaustausch nicht auch das NP-vollständige problem lösen müssen. Ihr Aufwand soll so gering wie möglich gehalten werden, wobei der Aufwand für einen Angreifer exponentiell oder faktoriell zur Sicherheit der Systems (z.B. die Länge des Schlüssels) ist.

Beispiele für solche Probleme sind der diskrete Logarithmus 1.2.1 und die Faktorisierung 1.2.2 eines Produkt von Primzahlen[6]. Weitere Beispiele wäre die Berechnung des Isomorphismus zweier Graphen, das Berechnen von Modularen Quadratwurzeln oder die Multiplikation auf elliptischen Kurven.

[5] Wikipedia 2021.
[6] Beutelspacher, Schwenk und Wolfenstetter 2015, S. 179.

Kapitel 2

RSA

RSA ist ein Algorithmus, welcher zur Klasse der Public-Key-Algorithmen gehört. Der Algorithmus wurde von R. Rivest, A. Shamir und L. Adleman erfunde und trägt deshalb ein Anagram Ihrer Namen.

2.1 Sicherheit von RSA

Die Sicherheit des RSA-Algorithmus basiert auf zwei mathematischen Problemen, welche unter Aufwand endlicher Ressourcen, nicht gelöst werden können. Hierbei wird sich sowohl auf RSA-gestützte Verschlüsselungs- und Signaturverfahren bezogen. Diese zwei Probleme sind:

- Faktorisierung einer bekannten Zahl, welche das Produkt zweier großer Primzahlen ist. Im Kontext von RSA ist diese Zahl n.
- Bestimmung des diskreten Logarithmus. Bei RSA wäre dies die Bestimmung von

$$d \mid m^d \equiv c \pmod{n}. \tag{2.1}$$

Für die Sicherheit der Public-Key-Verschlüsselung von RSA, spielt Unberechenbarkeit der Faktorisierung die Hauptrolle. Falls mit RSA signiert werden soll, ist zusätzlich die Unberechenbarkeit des diskreten Logarithmus wichtig. Ansonsten könnte der private und geheime Schlüssel abgeleitet werden.

Literatur

Beutelspacher, Albrecht, Jörg Schwenk und Klaus-Dieter Wolfenstetter [2015]. *Moderne Verfahren der Kryptographie*. springer [siehe S. 7, 8, 10–12].

Wikipedia [2021]. *Komplexitätstheorie* [siehe S. 12].