Einf.i.d.EDV /Techn.
Grundlagen der Telekomm.

Vorlesung WS 2009/10

Prof. Dr. Stefan Funke
(Informatik)
Universität Greifswald

stefan.funke@uni-greifswald.de

http://www.math-inf.uni-greifswald.de/informatik

Kurz zur Geschichte des Internets

- hervorgegangen aus dem Arpanet (1969), einem Projekt des DoD (Ziel: Vernetzung der für das DoD forschenden Universitäten und Forschungseinrichtungen)
- zu Beginn dominierte Email-Nutzung (sonst noch telnet, ftp, gopher)
- Durchbruch 1993 mit Entwicklung des Mosaic-Webbrowsers
- seit 1990 auch für nicht-universitäre Bereiche geöffnet
- heutzutage eine Vielzahl an Nutzungsarten: WWW, Peer-to-Peer, Online-Spiele, VoD, VoIP, ...
- vielen dieser Nutzungsarten ist das Internet durch seine eher traditionelle Struktur nicht mehr gewachsen

Technik

- das Internet ist sehr heterogen und besteht aus Firmennetzwerken, Providernetzwerken und Universitäts- und Forschungsnetzwerken
- physikalisch kommen hauptsächlich Glasfaserkabel als Backbones des Internets zum Einsatz
- logisch ist das Internet hierarchisch strukturiert (IP-Adressen, ...)
- ICANN (Internet Corporation for Assigned Names and Numbers) zuständig für Domain-Namen und IP-Adressen (untersteht dem US-Wirtschaftsministerium)

Einfluß auf die Gesellschaft

- hatte das Internet zu Beginn eher Nischencharakter, ersetzt es immer mehr konventionelle Medien (Zeitungen, Fernsehen, Radio, Telefon, Post)
- viele “Internet-Firmen”
- es vollzieht sich auch ein Wandel der Nutzung vom passiven “Websurfer” zum aktiven Gestalter – Stichwort Web 2.0, Online Communities, Soziale Netzwerke
- überarbeitete Gesetze und Rechtsprechung notwendig, um den veränderten Gegebenheiten Rechnung zu tragen (Privatsphäre, SPAM, ...)

Einige Fakten

- Anfang 2007 nutzten 1,13 Milliarden Menschen weltweit das Internet
- in Deutschland stehen in 75% aller Haushalte PCs mit Internetzugang
- der Energiebedarf des Internets in Deutschland entspricht der Leistung von 5 Kraftwerken mit je 1000 Megawatt

Grundlagen des Internets

- Wir werden im Folgenden einige Grundprimitive kennenlernen, ohne die das Internet und auch andere Netzwerke nicht realisierbar wären
- Zunächst werden wir uns mit Techniken der Datenübertragung beschäftigen, insbes.
  - Datenkompression
  - Fehlerkorrekturcodes
  - Verschlüsselung
- Später werden wir auf Konzepte wie IP-Adressen, Routing, etc. eingehen

Übertragung digitaler Daten

- Bandbreite ist nach wie vor eine knappe Resource => **Datenkompression**
- bei der Übertragung von Daten können Fehler auftreten, welche korrigiert werden müssen => **Fehlerkorrekturcodes**
- oft möchte man nicht, dass Dritte die übertragenen Daten mitlesen können => **Datenverschlüsselung**

Datenkompression

- in fast allen Bereichen ist Bandbreite eine sehr kostbare Ressource
- Beispiele:
  - Programmpakete werden meist z.B. mit WinZip komprimiert, um ihre Größe zu verringern
  - Musik kann in einem 10tel des ursprünglich notwendigen Platz gespeichert werden
  - Filme müssen komprimiert werden, um auf eine DVD/CD/Transponder zu passen

Datenkompression

- Man unterscheidet zwischen:
  - verlustfreier Kompression
    - ZIP
    - PNG
    - Bz2
    - ...
  - verlustbehafteter Kompression
    - MP3
    - MPEG, MPEG2 (DVD, DVB)
    - DivX
    - Ogg Vorbis
    - ...

Beispiele

- Ein 2stündiger Film auf DVD bräuchte unkomprimiert 720x576x3x25x60x60x2=ca. 208 GB !!!
- Auf einer DVD wird er auf ca. 5 GB komprimiert
- Per DVB-S wird er in ca. 2,5 GB übertragen
- Als DivX passt er auf eine CD (700 MB)
- Eine derart starke Kompression läßt sich nur mit Verlusten realisieren
- HD, FullHD nochmal eine andere Größenordnung ...

Datenkompression im Alltag (verlustfrei)

- Versenden von SMS (160 Zeichen)
- `Hi M.Zug versp.wg.Baust.Versuche asap da zu sein,werd aber sicher nicht gg 10 am Hbf sein.LG S.`
- Abkürzungen=Textkompression
- Funktioniert, weil der Empfänger mit einer Abkürzung dasselbe Wort assoziiert wie der Sender

Prinzip der Textkompression

- Sender und Empfänger haben ein gemeinsames "Wörterbuch"
- zu übertragende Nachricht wird als Aneinanderreihung von Verweisen in dieses Wörterbuch übermittelt
- Problem: was ist wenn ein solches gemeinsames Wörterbuch nicht existiert?

Gemeinsames Wörterbuch

- Als gemeinsames Wörterbuch kann der bislang übertragene Teil der Nachricht dienen:
- `Auch ein kleiner Beitrag ist ein Beitrag.`
- kann übermittelt werden als
- `Auch ein kleiner Beitrag ist 2 4.`
- Hierbei wird der bislang übermittelte Text als gemeinsames Wörterbuch genutzt und indiziert.

Gemeinsames Wörterbuch

- Bei sehr großen Texten würde man nur einen Teil – z.B. die letzten 1000 Wörter – des gesendeten Textes als Wörterbuch nutzen.
- `Auch ein kleiner Beitrag ist ein Beitrag.`
- kann dann übermittelt werden als
- `Auch ein kleiner Beitrag ist -4 -3.`
- Beachte: die Indexmarken sind Zahlen zwischen 1 und 1000 (bei einem Wörterbuch der Größe 1000).

Optimiertes gemeinsames Wörterbuch

- Beobachtung: Index “-1000” hat Größe 4, Index “-1” hat nur Größe 1
- Idee: Versuche Indizes so zu konstruieren, dass Worte die sehr oft vorkommen (z.B. “ein”) kleine Indizes haben, um noch mehr Platz zu sparen.
- Funktioniert bei einem Wörterbuch wie dem gezeigten nur implizit ...

Entropiekodierung

- Entropie=Maß für den Informationsgehalt
- Neues Verfahren:
  - Sender teilt Nachricht in Blöcke auf (z.B. 8KB)
  - S konstruiert für jeden Block ein Wörterbuch, in welchem Worte, die oft vorkommen, kurze Indizes/Codes besitzen
  - S überträgt jeweils das Wörterbuch gefolgt vom komprimierten Text an den Empfänger R
  - R benutzt das übertragene Wörterbuch, um den Text zu dekomprimieren

Einschub: Binärkodierungen

- Intern kann ein Rechner nur "0" und "1" repräsentieren (Binärzahlen)
- Die Zahl 65 stellt er z.B. binär dar als

  1000001

  was zu interpretieren ist als

  1x64+0x32+0x16+0x8+0x4+0x2+1x1
- Buchstaben können als Zahlen repräsentiert werden (z.B. ASCII-Code "A"=65, "B"=66,...)

Beispiel

- Wir ersetzen Wörter durch Buchstaben
- Nachricht: AAAABBCDAAA
- Binär kodiert hat diese Nachricht 22 Bits; Beispiel
- Alternatives Wörterbuch (binär):
  - `A=0`
  - `B=10`
  - `C=110`
  - `D=111`
- Komprimierte Nachricht:
  - `00001010110111000`
- wir brauchen nur noch 17 Bits!
- weiteres Beispiel mit diesem Wörterbuch

Entropiekodierung

Grundidee der Entropiekodierung:

Kodiere oft vorkommende Zeichen mit wenigen Bits, kodiere seltene Zeichen mit vielen Bits

Entropiekodierung

Beispiel: Ursprungsalphabet {A,B,C,D} wird “normalerweise” dargestellt durch {00,01,10,11}, d.h. ABACADA wird übertragen als 00010010001100 (14bits)

Man könnte nun jedoch {A,B,C,D} darstellen durch {0,10,110,111} und ABACADA übertragen als 010011001110 (12bits).

Entropiekodierung

Schlechtes Beispiel:

Man könnte {A,B,C,D} durch {0, 1, 10, 11} darstellen, d.h. ABACADA würde übertragen als 0101001110 (9bits).

Problem:

ACCADA würde durch den selben Binärstring dargestellt werden!

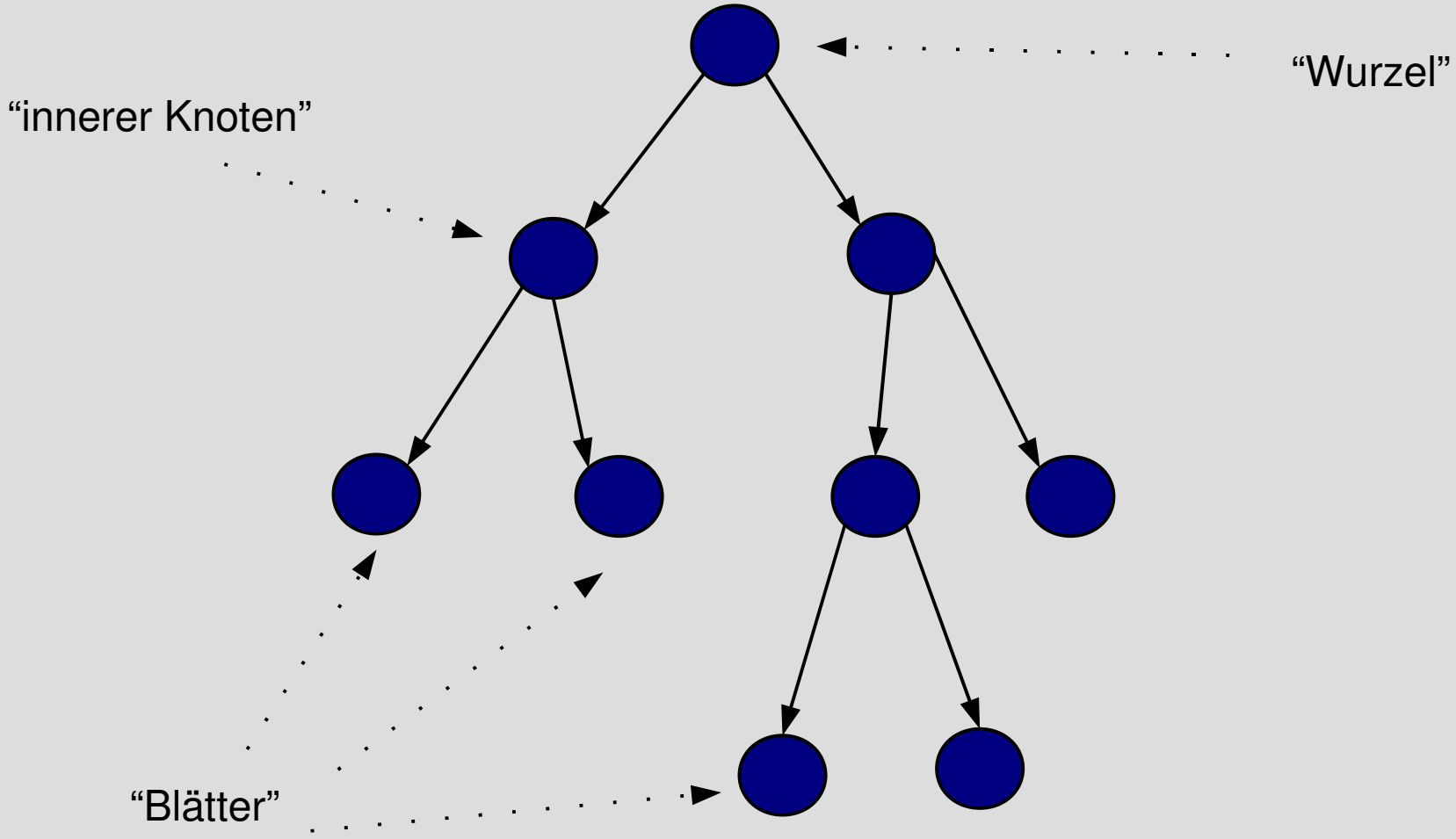
Präfixfreiheit

- Aus diesem Grund möchten wir ein Wörterbuch, bei dem kein Eintrag ein Präfix eines anderen ist:
- Bsp:
  - A=0, B=1, C=10, D=00 **falsch!**
  - A=1, B=01, C=001, D=000 **ok!**
  - A=0, B=10, C=11, D=110 **falsch!**
  - **...**

Shannon-Fano-Kodierung

- ... ist ein Verfahren, präfixfreie Codes (=Wörterbücher) zu erzeugen
- ... basiert auf der Darstellung binärer Codes durch einen Binärbaum

Shannon-Fano-Kodierung

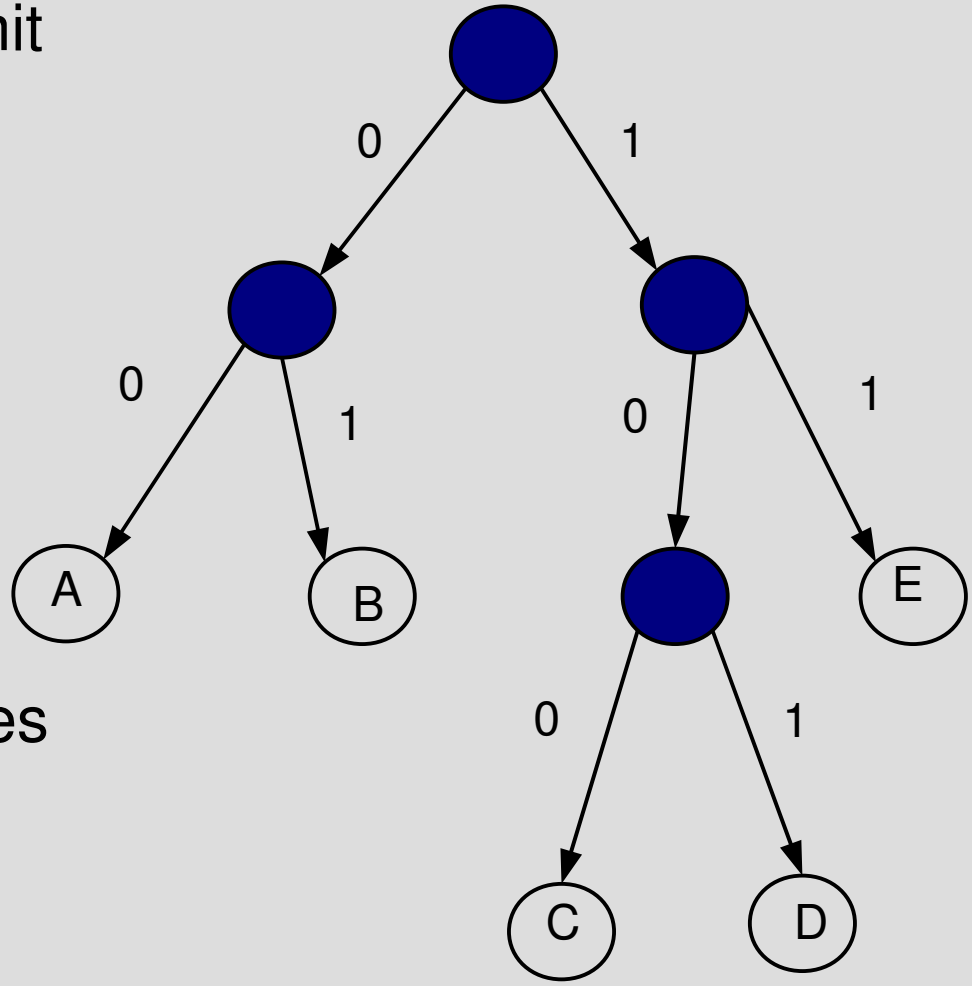


Shannon-Fano-Kodierung

- ein Binärbaum besteht aus einem **Wurzelknoten, inneren Knoten und Blattknoten**
- alle Knoten außer den Blattknoten haben maximal zwei Kindknoten
- alle Knoten außer dem Wurzelknoten haben genau einen Elterknoten
- Der Wurzelknoten hat keinen Elterknoten
- Beispiele ...

Shannon-Fano-Kodierung

- wir beschriften alle Linkskanten mit “0”, die Rechtskanten mit “1”
- wir schreiben die Wörter/ Buchstaben, die wir kodieren möchten, in die Blätter
- der Code für ein Wort ergibt sich aus den Beschriftungen entlang des Pfades von der Wurzel zum entsprechenden Blatt
- Beispiele...



Shannon-Fano-Kodierung

- Alle Codes, welche durch solche binären Bäume gegeben sind, sind präfixfrei (warum?)
- Dekodierung trivial mithilfe des Baums
- Beispiele ...

Shannon-Fano-Kodierung

- Annahme: wir haben einen Text gegeben, welcher aus 39 Zeichen besteht, und zwar aus 15 mal “A”, 7 mal “B”, 6 mal “C”, 6 mal “D” und 5 mal “E”
- Wir möchten einen Binärbaum konstruieren, in welchem die den Zeichen entsprechenden Blätter gemäß ihrer Häufigkeit im Baum auftauchen (häufige Zeichen weiter oben im Baum!)

Shannon-Fano-Algorithmus

1) Sortiere Zeichen gemäß ihrer Häufigkeit

ABCDE

2) Teile die Zeichen in dieser Reihenfolge in zwei Gruppen möglichst gleicher Größe

AB (22) CDE(17)

3) Konstruiere rekursiv für die beiden Zeichengruppen einen Binärbaum und verbinde Ihn mit einem Wurzelknoten

Shannon-Fano-Algorithmus

... Beispiele ...

Shannon-Fano-Algorithmus

Warum funktioniert das?

- durch die Sortierung landen Zeichen mit ähnlicher Häufigkeit im selben Teilbaum (und erhalten damit ähnliche Codelängen)

Meist recht gut, aber leider nicht immer optimal.

Huffman-Code-Algorithmus

- der Huffman-Code-Algorithmus berechnet ebenfalls einen Binärbaum, jedoch (beweisbar) optimal, d.h. die durchschnittliche Anzahl an Bits pro übertragenem Zeichen ist optimal!
- im Gegensatz zum Shannon-Fano-Algorithmus wird hier der Binärbaum von den Blättern aus konstruiert

Huffman-Code-Algorithmus

1. Erzeuge einen Wald von einzelnen Bäumen/Knoten (ein Knoten entspricht einem Zeichen; sein Gewicht der Anzahl an Vorkommen dieses Zeichens)
2. suche die beiden Bäume mit minimalem Gewicht (=Anzahl der durch die entsprechenden Bäume repräsentierten Zeichen) und vereinige sie zu einem Baum
3. wiederhole bis nur noch ein Baum übrig bleibt
4. Benutze den durch diesen Baum repräsentierten Code

Beispiele ...

Bemerkungen

- Shannon-Fano- und Huffman-Codes benötigen als Eingabe statistische Werte über die Häufigkeit von Zeichen/Worten
- oft ändern sich diese Häufigkeiten innerhalb eines Textkorpus; es gibt Verfahren, die entsprechenden Codes zu modifizieren, ohne sie komplett neu zu berechnen

Zusammenfassung Datenkompression

- Datenkomprimierung wichtig, um teuere Bandbreite effizient zu nutzen
- Verlustfreie vs. verlustbehaftete Komprimierung
- genauer betrachtet: verlustfreie Komprimierung von Text
- nicht genauer betrachtet: verlustbehaftete Komprimierung (Ton, Bild – MP3, DiVX); relativ kompliziert+mathematisch

Fehlerkorrekturcodes

- bei der Übertragung von Daten können Störungen auftreten, welche die Daten verfälschen
- man braucht Verfahren, die erkennen, ob eine Störung vorgelegen hat, und dadurch entstandene Verfälschungen der Daten korrigieren können

Fehlerkorrekturcodes im Alltag

Bsp.: Sie schreiben sich eine Telefonnummer auf einen Schmierzettel:

03834-86-4642

Sie könnten dieselbe Information jedoch auch aufschreiben als

03834-86-4642

Fehlerkorrekturcodes im Alltag

- Vom Informationsgehalt sind beide Notizen äquivalent
- die erste Notiz ist jedoch robuster hinsichtlich Störungen von außerhalb, z.B. Soßenflecken

Fehlerkorrekturcodes

- Fehlerkorrekturcodes sind Verfahren, Daten robuster gegen Störungen zu machen
- sie bewirken in gewisser Weise das Gegenteil von dem, was Datenkompression erreicht
- Datenkompression versucht Redundanz zu minimieren
- Fehlerkorrekturcodes versuchen Redundanz zum Zwecke der Robustheit gegen Störungen einzusetzen

Parity Checks

- Nehmen wir an, Sie möchten folgende binäre Nachricht übertragen

  `01101001`

  und sichergehen, dass keine Bits unterwegs geflippt worden sind
- Idee: wir übertragen ein zusätzliches Bit, welches die Korrektheit der Übertragung zertifizieren soll

Parity Checks

- Wir zählen die Anzahl der 1'en in der Nachricht und hängen ein Paritätsbit an (1 falls die Anzahl ungerade ist, 0 andernfalls):

  `01101001`**0**
- anders ausgedrückt: die Summe der 1en soll immer gerade sein (Datenbits und Paritybit)
- der Empfänger der Nachricht kann nun durch nachzählen der 1en überprüfen, ob die Nachricht unverfälscht verschickt wurde
- Beispiele ...

Parity Checks

- das Parity Check Bit kann nur ein geflipptes/ gestörtes Bit erkennen
- eine Korrektur der Nachricht ist leider nicht möglich; typischerweise wird die Nachricht einfach neu angefordert

Repetition Codes

- ein sehr einfaches Beispiel eines Error Correcting Codes ist der n-Repetitioncode
- Idee für den 3-Repetitioncode: wenn ich eine 0 übertragen möchte, übertrage ich einfach 000, bei einer 1 übertrage ich 111
- In jedem übertragenen 3er Block kann ein Bit falsch gesetzt sein, dennoch kann die Nachricht rekonstruiert werden
- 2 Bitfehler können erkannt, nicht repariert werden
- Beispiele ...

Hamming Codes

- Repetition Codes sind aufgrund ihrer naiven Konstruktion nicht besonders effektiv
- um wenige Bitfehler zu erkennen, wird die Nachrichtenlänge um ein Vielfaches aufgebläht
- Hamming Codes sind ein effizienteres Verfahren, error correcting Codes zu konstruieren

Begriff: Hamming Distanz

- Die Hamming Distanz ist ein Maß, um Strings (Zeichenketten) – sowohl binär, als auch "normale" – zu vergleichen
- Die Hamming Distanz zweier Worte von gleicher Länge entspricht der Anzahl der Stellen, an welchen die beiden Worte differieren
- Beispiele ...

Hamming Distanz von Codes

- da wir uns für Codes (Zuordnungen/Wörterbücher) interessieren, möchten wir auch die Hamming Distanz eines Codes bezeichnen
- im Folgenden betrachten wir nur Codes mit Codewörtern fixer Länge
- die Hamming Distanz eines Codes ist die minimale Hamming Distanz zweier Codewörter
- alles Codes haben Hamming Distanz>0
- Beispiele ...

Warum interessiert uns die Hamming Distanz eines Codes?

- Idee: wenn unser Code eine große Hamming Distanz hat, kann ein einzelnes Codewort durch Übertragungsfehler relativ stark gestört werden aber trotzdem noch erkennbar sein
- man muß notwendigerweise einen Kompromiss zwischen Codewortgröße und Hamming Distanz eingehen
- Beispiele ...

Warum interessiert uns die Hamming Distanz eines Codes?

- Wenn wir einen Code mit Hamming Distanz k hätten, könnten wir
  - < k Fehler erkennen
  - < k/2 (abgerundet) Fehler korrigieren
- d.h. für Korrektur brauchen wir einen Code mit Hammingdistanz mindestens 3
- Beispiele ...

Hamming Codes

- Notation: ein **(N,d) Code** bezeichnet einen Code, in dem ein Wort aus N Bits besteht, wovon d Datenbits sind und k=N-d Korrekturbits
- Beispiel Paritybitcode:N=9 Bits, von denen d=8 Datenbits waren und k=1 Korrekturbit
- Was fuer ein (N,d) Code ist der Repetition Code?
- Die Informationsrate ist das Verhältnis zwischen Datenbits und Gesamtbitanzahl, hier 8/9 bzw. 1/3
- **Ziel: Code maximaler Hamming Distanz bei hoher Informationsrate**

Hamming Codes - Algorithmus

- alle Bits, welche Potenzen von 2 sind, sind Korrekturbits (d.h. Positionen 1, 2, 4, 8, ...)
- die restlichen Bits tragen die eigentl. Daten
- das i-te Korrekturbit enthält die Parität der Positionen, welche Bit i in ihrer Binärrepräsentation gesetzt haben

Hamming Codes - Algorithmus

```
p1 p2 p3 p4 p5 p6 p7
c1 c2 d1 c3 d4 d5 d6
```

- c1 enthält die Parität von p1 p3 p5 p7
- c2 enthält die Parität von p2 p3 p6 p7
- c3 enthält die Parität von p4 p5 p6 p7

Beispiel: Alle 16 4-Bit-Datenwörter samt Paritybits für (7,4) Code

(7,4)-Hamming Codes

000**0**000	**11**1**0**000
110**1**001	**00**1**1**001
010**1**010	**10**1**1**010
100**0**011	**01**1**0**011
100**1**100	**01**1**1**100
010**0**101	**10**1**0**101
110**0**110	**00**1**0**110
000**1**111	**11**1**1**111

Hamming Code

- Keine zwei Codewörter haben Hammingdistanz <3
- d.h. man kann zwei fehlerhafte Bits erkennen und ein fehlerhaftes Bit korrigieren
- Beispiele ...

Hamming Code

- Hamming Codes werden nicht nur in der Telekommunikation eingesetzt, sondern auch rechnerintern z.B. bei der Ansteuerung des Arbeitsspeichers
- in der Praxis werden nicht (7,4) Codes eingesetzt, sondern eher (63,57) Codes wegen der besseren Informationsrate

Datenverschlüsselung

- Sobald Sie sich im Internet bewegen, hinterlassen Sie gewollt oder ungewollt Daten
- Manche dieser Daten (z.B. Kreditkartennummern, Bankverbindungen, persönliche Emails, ... sollten nicht in fremde Hände gelangen
- auch möchten Sie vielleicht, dass nicht jedermann weiß, welche Webseiten Sie besuchen

Datenverschlüsselung

Wir werden Verfahren kennenlernen, welche

- es erlauben, Daten geschützt zu übertragen, sodass kein Dritter mitlesen kann
- es dem Kommunikationspartner erlauben, seine Identität zu “beweisen”
- es erlauben, digitale Dokumente zu signieren
- es erlauben, sich anonym im Internet zu bewegen

Datenverschlüsselung: Szenario 1

Nehmen Sie an, Sie möchten Ihrem Freund eine Nachricht schicken und sichergehen, daß niemand außer Ihrem Freund die Nachricht lesen kann.

Datenverschlüsselung: Szenario 2

Sie möchten eine Überweisung mittels Online-Banking tätigen. Wie können Sie sichergehen, dass die Webseite, auf der Sie sich mit Ihrem Passwort einloggen auch wirklich die Ihrer Bank ist?

Datenverschlüsselung: Szenario 3

Sie bekommen eine Email mit brisantem Inhalt von Ihrem Freund. Wie können Sie sicher sein, dass die Email wirklich von Ihrem Freund stammt und nicht von jemandem, der sich als Ihr Freund ausgibt?

Datenverschlüsselung: Szenario 4

Sie befinden sich in einem Land, in dem es um die Meinungsfreiheit nicht sonderlich gut bestellt ist. Wie können Sie dennoch Webseiten besuchen und Beiträge zu aktuellen politischen Themen verfassen, ohne dass sie dadurch in Schwierigkeiten geraten?

Grundsätzliches zum Thema Datenverschlüsselung

Ein Verschlüsselungsverfahren ist nicht deshalb gut/sicher, weil niemand das Verfahren genau kennt und es deshalb auf den ersten Blick schwierig ist, es zu knacken.

... im Gegenteil

Grundsätzliches zum Thema Datenverschlüsselung

Ein Verschlüsselungsverfahren ist gerade dann als sicher einzuschätzen, wenn jeder leicht versteht, wie es funktioniert, aber es trotzdem schwierig ist, damit verschlüsselte Nachrichten zu entschlüsseln (ohne den entsprechenden Schlüssel).

Grundsätzliches zum Thema Datenverschlüsselung

- es gibt nur wenige Verschlüsselungsverfahren, die beweisbar “unknackbar” sind.
- für die meisten anderen Verfahren möchte man dennoch irgendwie zeigen, dass sie schwer zu überwinden sind
- in der Mathematik/Informatik wendet man hierzu “Reduktionen” an

Reduktionen

- es gibt eine Anzahl “anerkannt” schwieriger Probleme in der Mathematik/Informatik
- Beispiele: Travelling Salesman Problem, Primfaktorzerlegung
- man versucht nun, zu überprüfende Verschlüsselungsverfahren auf solche anerkannt schwierige Probleme zu “reduzieren”, d.h.

Reduktion - Beispiel

- angenommen, man könnte das neue Verschlüsselungsverfahren “TopCrypt” schnell knacken, dann könnte ich auch eine sehr große Zahl in Primfaktoren zerlegen
- d.h. für jede große Zahl kann ich eine mit TopCrypt verschlüsselte Nachricht erzeugen, sodass das Entschlüsseln dieser Nachricht mir (implizit) auch eine Primfaktorzerlegung dieser Zahl liefert

Grundsätzliches zum Thema Datenverschlüsselung

- nur Verfahren, welche in dieser Art und Weise auf ihre Sicherheit geprüft wurden, sollten als sicher angesehen werden
- ist das Verfahren einfach nur “kompliziert”, heißt nicht, dass es sicher ist; bei entsprechend starken Interessen wird es typischerweise auch über kurz oder lang geknackt

Steganographie

- manchmal ist es nicht erwünscht, dass Dritte erfahren, dass überhaupt eine (verschlüsselte) Nachricht übertragen wird
- Steganographie = “Verstecken” von Nachrichten in anderen Daten (z.B. Bildern, Videos, ...)

Perfekte Verschlüsselung - OneTimePads

- die einzige Art und Weise, Nachrichten zu verschlüsseln, sodass es absolut unmöglich ist, für einen Angreifer, die Nachricht mitzulesen
- Szenario 1: wir möchten einem Freund eine Nachricht schicken und sichergehen, dass niemand die Nachricht mitlesen kann

Perfekte Verschlüsselung - OneTimePads

- die Nachricht ist in digitaler Form eine Sequenz von 0en und 1en einer bestimmten Länge 0100101010101101010101010011011000
- Annahme: Sie haben sich zuvor mit Ihrem Freund getroffen und ein OneTimePad ausgetauscht

Perfekte Verschlüsselung - OneTimePads

- ein OneTimePad ist eine zufällige Sequenz von 0en und 1en (vorzugsweise länger als die später auszutauschende Nachricht): 0101011000110111010101110101000010101
- nach dem Treffen haben Sie und Ihr Freund das gleiche OneTimePad zuhause

Perfekte Verschlüsselung - OneTimePads

- bevor Sie die Nachricht 01001010101011010101010010011000 ihrem Freund schicken, kombinieren Sie diese mit Ihrem OneTimePad: 01010110001101111010101110101000101 01
- ... und zwar mit der XOR-Operation
- ... dann schicken Sie die Nachricht an Ihren Freund

Perfekte Verschlüsselung – OneTimePads und XOR

- XOR ist eine Verknüpfungsregel, welche 2 binäre Ziffern (also 0 oder 1) zu einer dritten binären Ziffer verknüpft, und zwar wie folgt:
- 0 XOR 0 = 0
  0 XOR 1 = 1
  1 XOR 0 = 1
  1 XOR 1 = 0
- Längere Ziffernfolgen “verXORen” wir Ziffer für Ziffer ... Beispiele

Perfekte Verschlüsselung – OneTimePads und XOR

- angenommen Sie haben Nachricht A mit OTP B verXORt und Sequenz C erhalten
- dann ergibt C XOR B wieder die Nachricht A
- Ihr Freund kann also die von Ihnen gesendete Nachricht durch verXORen mit dem OTP entschlüsseln
- Beispiele ...

Perfekte Verschlüsselung – OneTimePads

- warum ist diese Verschlüsselung perfekt im Sinne von “unknackbar” ?
- der OTP ist zufällig generiert, d.h. wenn ich alle Ziffern bis auf eine des OTP kenne, hilft mir das nicht, die fehlende Ziffer vorherzusagen
- dasselbe gilt für C; d.h. die verschlüsselte Nachricht C verhält sich genauso wie eine zufällige Sequenz von 0en und 1en
- es ist also unmöglich, aus C Information bzgl. der ursprünglichen Nachricht A zu lernen

Perfekte Verschlüsselung – OneTimePads - Caveats

- wie der Name schon sagt, darf ein OneTimePad nur einmal verwendet werden
- bei mehrfacher Verwendung können durch Statistiken über die in der Originalnachricht verwendeten Sprache Ziffern vorhergesagt werden
- bei hinreichend vielen mit diesem einen OTP verschlüsselten Nachrichten, können der OTP und damit auch alle Nachrichten entschlüsselt werden

Perfekte Verschlüsselung – OneTimePads - Caveats

- wirklich zufällige Ziffern zu generieren ist nicht einfach
- der in Betriebssystemen integrierte Zufallszahlengenerator reicht oft nicht aus
- physikalische Messungen können relativ gute zufällige Zahlen generieren
- bei zu schwacher Zufälligkeit ist das System angreifbar

Perfekte Verschlüsselung – OneTimePads

- in der Praxis werden OTPs eher selten, und nur in besonders sensiblen Bereichen eingesetzt, da die physische Übergabe der OTPs meist zu aufwändig ist

Asymmetrische Verschlüsselung

- bei OneTimePads wie auch symmetrischen Verschlüsselungen (später dazu mehr) gibt es einen “Schlüssel”, der sowohl für die **Ver**schlüsselung, als auch für die **Ent**schlüsselung verwendet wird
- dies impliziert, dass dieser Schlüssel unter allen Umständen geheim gehalten werden muss, und nur dem Sender und dem Empfänger bekannt sein darf

Asymmetrische Verschlüsselung

- bei der asymmetrischen Verschlüsselung gibt es zwei Schlüssel
  - einen öffentlichen Schlüssel, der jedermann zugänglich gemacht werden kann
  - einen privaten Schlüssel, auf den sein Inhaber Zugriff haben sollte
- Kommunikationspartner müssen keinen gemeinsamen Schlüssel kennen
- Public Key Verfahren gehören zur Klasse der asymmetrischen Verschlüsselungen

Szenario 1

- Nehmen Sie an, Sie möchten Ihrem Freund eine Nachricht schicken und sichergehen, daß niemand außer Ihrem Freund die Nachricht lesen kann.

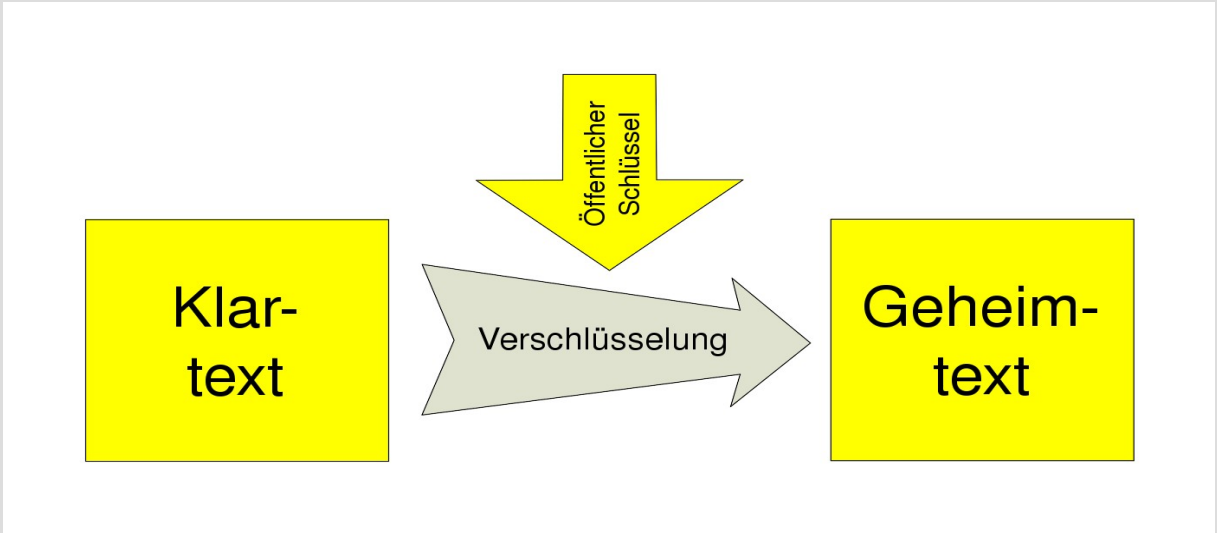
Szenario 1

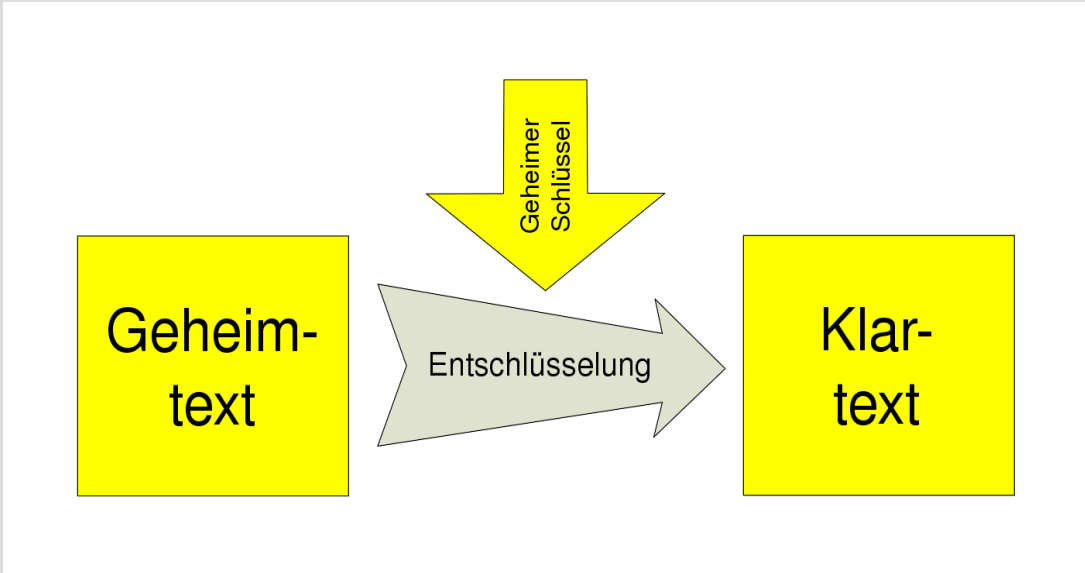
- ihr Freund generiert ein Paar von Schlüsseln (Ö,P), einen öffentlichen Schlüssel Ö und einen privaten Schlüssel P
- den öffentlichen Schlüssel Ö stellt er z.B. auf seine Webseite
- den privaten Schlüssel P hält er geheim

Szenario 1

- Sie laden den öffentlichen Schlüssel Ö Ihres Freundes runter und verschlüsseln damit die Nachricht an Ihren Freund
- die verschlüsselte Nachricht wird an Ihren Freund verschickt
- Ihr Freund benutzt seinen privaten Schlüssel, um die Nachricht zu entschlüsseln

Asymmetrische Verschlüsselung





(Diagramme aus Wikipedia)

Asymmetrische Verschlüsselung

- Vorteil: es muß kein gemeinsamer Schlüssel auf “sicherem” Weg übertragen/übergeben werden
- Nachteil: die meisten asymmetrischen Verschlüsselungsverfahren sind im Vergleich zu symmetrischen Verfahren relativ langsam (ca. Faktor 1000)

Asymmetrische Verschlüsselung: der RSA Algorithmus

- der öffentliche Schlüssel Ö ist ein Zahlenpaar (e,N)
- der private Schlüssel P ist ein Zahlenpaar (d,N)
- N nennt sich RSA-Modul
- e ist der Verschlüsselungsexponent
- d ist der Entschlüsselungsexponent

Asymmetrische Verschlüsselung: der RSA Algorithmus

- e,d und N werden wie folgt erzeugt:
  - wähle zwei zufällige (große) Primzahlen p,q
  - setze $N:=p \cdot q$
  - berechne $F(N)=(p-1) \cdot (q-1)$ (Eulersche Phi-Funktion)
  - wähle eine Zahl $1<e<F(N)$, welche teilerfremd zu F(N) ist
  - d:= multiplikativ Inverses zu e bzgl. des Modulus F(N), d.h. $e \cdot d=1 \pmod{F(N)}$

RSA-Algorithmus: Beispiel

- p=11, q=13
- RSA-Modul N=11·13=143
- F(N)=10·12=120
- suche e teilerfremd zu 120, z.B. e=23
- (e,N)=(23,143) ist der öffentliche Schlüssel
- mithilfe des euklidschen Algorithmus finden wir d=47, d.h. (d,N)=(47,143) ist der private Schlüssel

RSA-Algorithmus: Verschlüsselung

- zur Verschlüsselung einer Nachricht K, wendet man folgende Formel an:
  $C=K^e \bmod N$
- C ist die verschlüsselte Nachricht

RSA-Algorithmus: Verschlüsselung

- zu verschlüsselnde Nachricht K=7
  $C=7^{e} \bmod N=7^{23} \bmod 143=2$
- C=2 ist die verschlüsselte Nachricht und wird übertragen

RSA-Algorithmus: Entschlüsselung

- zur Entschlüsselung einer verschlüsselten Nachricht C, wendet man folgende Formel an:
  $K=C^d \bmod N$
- K ist die entschlüsselte Nachricht

RSA-Algorithmus: Entschlüsselung

- zu entschlüsselnde Nachricht C=2
  $K=2^d \bmod N=2^{47} \bmod 143=7$
- K=7 ist die entschlüsselte Originalnachricht

Datenverschlüsselung: Szenario 3

Sie bekommen eine Email mit brisantem Inhalt von Ihrem Freund. Wie können Sie sicher sein, dass die Email wirklich von Ihrem Freund stammt und nicht von jemand, der sich als Ihr Freund ausgibt?

Signatur von Nachrichten

- RSA Verschlüsselung hilft auch in Szenario 3
- Ihr Freund schickt Ihnen nicht nur die Originalnachricht K, sondern auch eine verschlüsselte Version C, welche er mit seinem privaten(!) Schlüssel verschlüsselt
- $C=K^d \bmod N$

Signatur von Nachrichten

- als Empfänger der Nachricht, können Sie nun einfach kontrollieren, ob die verschlüsselte Nachricht zur unverschlüsselten “passt”
- Sie benutzen den öffentlichen Schlüssel e Ihres Freundes und berechnen
- $K' = C^e \bmod N$
- falls K'=K, kam die Nachricht wirklich von Ihrem Freund

Längeres Beispiel

- wir wollen den Text WIKIPEDIA verschlüsselt übertragen
- Annahme: wir können Buchstaben durch Zahlen repräsentieren A=01, B=02, ...
- d.h. WIK IPE DIA=230911 091605 040901
- durch Wahl von p,q,N haben wir erzeugt: (e,N)=(1721,263713), (d,N)=(1373,263713)

Längeres Beispiel

- wir verschlüsseln immer Sequenzen von 6 Ziffern mithilfe des RSA Algorithmus
- K=K1 K2 K3=230911 091605 040901
- $C1=K1^{e} \bmod N$, C2= ....
- C=C1 C2 C3
- entschlüsseln erfolgt analog

RSA: Sicherheit

- die Entschlüsselung einer mit RSA verschlüsselten Nachricht ist (bei bekanntem e,N) nicht leichter als das Finden des privaten Schlüssels d
- das Herausfinden des privaten Schlüssels d (bei bekanntem e und N) ist genauso schwer wie die Primfaktorzerlegung
- es wird seit langem vermutet, dass Primfaktorzerlegung sehr schwierig ist
- es gibt aktuell keine praktikablen Verfahren, große Zahlen effizient zu faktorisieren

RSA: Vor-/Nachteile

- Vorteil: es muß kein gemeinsamer Schlüssel übergeben werden
- Nachteil: wer stellt sicher, dass der (z.B. auf der Webseite) veröffentlichte Schlüssel wirklich der öffentliche Schlüssel des Kommunikationspartners ist?
- Nachteil: Ver-/Entschlüsselung sehr langsam

Zertifizierung eines Schlüssels

- Sie möchten sichergehen, dass die Email-Adresse/der Public Key Ihres Freundes auch wirklich Ihrem Freund gehört
- dafür muss Ihr Freund seine Email-Adresse als auch seinen PK durch eine dritte Partei (denen Sie vertrauen) signieren lassen (normalerweise unter Vorlage von Personalausweis)
- Bsp.: c't Krypto-Kampagne

Hybride Verschlüsselung

- RSA ist zu langsam für große Nachrichten
- in der Praxis wird folgendes angewandt:
  - der Sender erzeugt einen zufälligen Schlüssel S eines symmetrischen Verschlüsselungsverfahrens (ähnlich OTP)
  - er verschlüsselt S mit dem öffentlichen Schlüssel des Empfängers zu S' und die Nachricht K mit S zu K'
  - er schickt S' und K' an den Empfänger

Hybride Verschlüsselung

- was macht der Empfänger?
  - er benutzt seinen privaten Schlüssel um aus S' den zufällig generierten Schlüssel S des symmetrischen Verfahrens zu erzeugen
  - er benutzt S um aus K' den Klartext K zu gewinnen

Symmetrische Verschlüsselungsverfahren

- OTPs haben den Nachteil, dass das OTP so lang sein musste, wie die zu verschlüsselnde Nachricht
- es gibt andere Verfahren, welche nur einen kürzeren Schlüssel (< 1024 bit) benötigen, und ebenfalls zur symmetrischen Verschlüsselung geeignet sind
- sie sind theoretisch nicht so sicher wie OTP, werden jedoch praktisch als sicher eingestuft
- Beispiele: 3DES, AES

Datenverschlüsselung: Szenario 2

Sie möchten eine Überweisung mittels Online-Banking tätigen. Wie können Sie sichergehen, dass die Webseite, auf der Sie sich mit Ihrem Passwort einloggen auch wirklich die Ihrer Bank ist?

... kann man ebenfalls durch PK Verschlüsselung erreichen (selber überlegen!)

Anonymität im Internet

- wenn Sie sich im Internet bewegen, hinterlassen Sie Spuren, die es anderen erlaubt, festzustellen, welche Webseiten Sie besucht haben und ggf. sogar, welche Daten Sie übermittelt haben
- das ist oft nicht gewünscht, insbesondere in Ländern, in denen freie Meinungsäußerung nicht gern gesehen wird

Crashkurs Internet

- das Internet ist ein logisch hierarchisch organisiertes Netzwerk
- jedem Rechner ist eine eindeutige sogenannte IP-Adresse zugeordnet
- Beispiel: 141.53.34.90
- eine IP-Adresse kann statisch vergeben werden (Rechner hat immer dieselbe IP-Adr.) oder auch dynamisch (z.B. bei Einwahl über Internet-Provider)

Crashkurs Internet

- typischerweise haben Sie nichts mit IP-Adressen zu tun, sondern meist mit Adressen der Form

  www.google.de
- es gibt den “Domain Lookup Service” (DNS), welcher Adressen in obiger Form in die entsprechenden IP-Adressen übersetzt

Beispiel: Aufruf der Webseite www.google.de

1) “www.google.de” wird an Ihren DNS Server (dessen IP-Adresse Sie normalerweise kennen - 141.53.9.12) geschickt (in einem Paket zusammen mit Ihrer eigenen IP-Adresse - 141.53.34.90, damit er auch weiß, an wen er die Antwort schicken muß)
2) nach einiger Zeit bekommen Sie als Antwort die IP-Adresse von www.google.de vom DNS Server – 66.249.91.103
3) nun schicken Sie ein Anfrage an 66.249.91.103, mit der bitte um Antwort mit dem HTML-Code der Startseite von Google (wieder mit Ihrer eigenen IP-Adresse für die Antwort)
4) nach einiger Zeit bekommen Sie von 66.249.91.103 ein Paket mit den gewünschten Daten

Wie werden Nachrichten verschickt/geroutet?

- die hierarchische Struktur der vergebenen IP-Adressen läßt sich gut mithilfe eines Baumes verdeutlichen
- die Blätter dieses Baumes entsprechen den Rechnern/Teilnehmern
- die inneren Knoten des Baumes entsprechen Routern, Teilbäume oft geographisch eingegrenzten Regionen
- Beispiel ... Baum
- Beispiel .... dnstools

Anonymität(?)

- der DNS Server bekommt für jede aufgerufene Webseite eine entsprechende Anfrage (inkl. Ihrer IP-Adresse)
- alle Router auf dem Weg zur gewünschten Webseiten-IP-Adresse kennen sowohl Ziel-IP als auch Quell-IP (Sie!)
- falls Sie Ihre Daten nicht verschlüsseln, können auch alle mitlesen

Anonymität(?)

- ab 2009 sind Internetprovider verpflichtet, Zugangsdaten 6 Monate lang zu speichern:
  - welche IP-Adresse für welchen Zeitraum zugeteilt wurde
  - Quell- und Ziel-IP aller verschickten Emails(?)
  - Handy ab 2008: Quell-/Zielnummer, Ort, Dauer
- d.h. es kann bis zu 6 Monate später ermittelt werden, welche Webseiten Sie besucht und wem Sie Emails geschickt haben ...

Anonymität(?)

- Sie können davon ausgehen, dass viele Webseitenanbieter Ihre IP-Adresse bei Zugriffen auf deren Webseite speichern (und dadurch auch Hinweise auf Ihre Aufenthaltsorte erlangen)
- zusammen mit Informationen, die Ihr Browser übermittelt, geben Sie ziemlich viel Informationen über sich preis

Browserspezifische Informationen

- Heutige Internet-Browser schicken (außer IP-Adresse) noch weitere Informationen mit, z.B. die vor der aufgerufenen Seite zuletzt besuchte Seite (Referer-URL)
- Browser-Typ, Betriebssystem, eingestellte Sprache, etc.
- viele Webseiten hinterlassen auch sog. Cookies, die es erlauben, Sie bei wiederholtem Besuch wiederzuerkennen
- Beispiel ... methodica

Anonymität via Proxies

- eine Möglichkeit, etwas Anonymität zu gewährleisten ist der Einsatz von Proxies
- ein Proxy-Rechner agiert als “Mittelsmann” für alle Ihre Internetverbindungen
- das heißt ...

Anonymität via Proxies

1) Sie schicken die Adresse “www.google.de” verschlüsselt an Ihren Proxy
2) Ihr Proxy führt (wie Sie zuvor) die Anfrage der Webseite (auch die DNS Anfrage) durch und schickt Ihnen das Ergebnis in einem verschlüsselten Paket

Anonymität via Proxies

- ein Dritter kann nun nicht mehr ohne weiteres feststellen, welche Webseiten Sie besucht haben
- bei Überwachung des Proxies (und wenig Verkehr darüber) ist die Sicherheit jedoch nicht sonderlich hoch
- Beispiel ...

Anonymität via Proxies

Ihre Anonymität hängt nun davon ab, ob Sie dem Proxybetreiber vertrauen (sehr wichtig) und dass dieser nicht überwacht wird (und entsprechende Vorkehrungen bei Ihrem Browser).

Vorsicht:

· Erweiterungen im Browser wie z.B. ActiveX-Plugins, Flash, Java, ... umgehen möglicherweise einen eingerichteten Proxy

Anonymität via Proxies

- einzelne Anonymizer-Proxies sind naheliegende Ziele für interessierte Parteien
- man kann es entsprechenden Interessenten schwerer machen, indem man ein Netzwerk von Proxies installiert (möglichst auch nicht auf ein einzelnes Land beschränkt)

Anonymität via Proxies

- vor jeder Anfrage ins Internet wählt man eine zufällige Route in diesem Netzwerk von Proxies aus, und führt die Anfrage über diese Route
- ein Angreifer müßte nun alle Proxies des Netzwerkes überwachen, um Informationen über Ihre Seitenzugriffe zu erhalten

Anonymität via Proxies

- der Verkehr zwischen den Proxies läuft verschlüsselt ab
- sensible Daten sollten dennoch verschlüsselt werden, da sonst die Daten vom letzten Proxy (Exit-Node) zum eigentlichen Ziel unverschlüsselt übertragen werden
- je mehr Nutzer das Anonymisierungsnetzwerk nutzen, desto anonymer ist es

Gängige Anonymizer

- JAP (Java Anon Proxy)/JonDo
  anon.inf.tu-dresden.de
  ... ein Netzwerk von statischen Mixkaskaden ...
  zu Beginn kostenlos nutzbar, im Moment Übergang zu einem Bezahlservice
  anonymisiert im Moment nur Webtraffic
- TOR (The Onion Router)
  ... Netzwerk von dynamischen Mixkaskaden
  ... anonymisiert auch InstantMessaging, SSH, P2P, Email ...
- Beispiel ...

Netzwerke

- Netzwerke bestehen aus Knoten (Rechner/Router) und sie verbindende Kommunikationskanäle
- Sie werden charakterisiert durch
  - ihre Topologie, d.h. nach welchem Muster die Knoten miteinander verbunden sind
  - die Art des Routing der Nachrichten im Netzwerk
  - die Übertragungstechnologie (drahtlos, verdrahtet, ...)

Netzwerktopologien

- Gängige Netzwerktopologien:
  - all-to-all
  - Sterntopologie
  - Baumtopologie
  - Ringtopologie
  - Meshtopologie
  - ... Mischungen davon
- Beispiele ... Vor-/Nachteile?
- die Netzwerktopologie hat unmittelbaren Einfluß auf das Routing im Netzwerk

Netzwerktopologien

- Wann ist eine Netzwerktopologie gut/schlecht?
  - eine gute NT sollte für jedes Paar von Knoten mehrere Pfade zum Datenaustausch erlauben (Robustheit gegen Störungen)
  - eine gute NT sollte es Routingschemata erlauben, die Last im Netzwerk zu verteilen
  - eine gute NT sollte möglichst wenig Links enthalten (Kostenfrage!)

Routing in Netzwerken

- Routing=die Art und Weise, wie Nachrichten von einem Rechner zu einem bestimmten anderen Rechner verschickt werden
- manche Netzwerktopologien erlauben keine Freiheit beim Routing (welche?)
- Unterscheidung in
  - zentrale Routingschemata
  - dezentrale Routingschemata

Routing in Netzwerken

- Wann ist ein Routingschema gut/schlecht?
  - gute Routingschemata sollten die Last im Netzwerk möglichst gleichmäßig verteilen
  - gute Routingschemata sollten sich an Veränderungen im Netzwerk anpassen können
  - der Overhead bei der “Planung” der Route sollte gering im Vgl. zur Menge der verschickten Daten sein

Zentrales/Dezentrales Routing

- Analogien: Straßen- bzw. Schienenverkehr
- Schienenverkehr:
  - Züge fahren auf dem Schienennetz gemäß eines zentral im Voraus geplanten Fahrplans
  - falls keine Störungen auftreten, ist der Zustand des “Netzwerkes” zu jedem Zeitpunkt bekannt
- Straßenverkehr:
  - Autofahrer planen ihre Route individuell (nach Erfahrung/Verkehrsvorhersage)
  - Staus und “Lastverteilung” schwer vorhersagbar

Zentrales/Dezentrales Routing Vor-/Nachteile

- Zentrales Routing:
  - globales Wissen erlaubt es, optimal die Nachrichten/Waren zu verschicken (+)
  - Änderungen der Rahmenbedingungen bedingen Neuplanung (-)
  - Neuplanung kann bei entsprechend großem Netzwerk sehr schwierig sein – schlechte Skalierbarkeit (-)
  - zentrale Planung oft nicht durchsetzbar -- “selfish agents” (-)

Zentrales/Dezentrales Routing Vor-/Nachteile

- Dezentrales Routing:
  - Planung erfolgt individuell; es sind keine globalen Informationen nötig (+)
  - Änderungen der Rahmenbedingungen einfach zu berücksichtigenbedingen (+)
  - gute Skalierbarkeit (+)
  - globale Qualitätsmerkmale schwierig zu berücksichtigen (-)
  - unklar, wie z.B. Lastbalancierung erreicht werden kann (-)

“Price of Anarchy”

- man kann versuchen, zu quantifizieren, wieviel an “Gemeinwohl” man verliert, wenn man dezentrales Routing betreibt
- in unserem Beispiel: bei gleichem Netzwerk planen wir zum einen der Verkehr zentral (wie die Bahn) oder lassen jedes Individuum seien eigene Entscheidung treffen
- wird auch “Price of Anarchy” genannt

“Price of Anarchy”

- Beispiel:
  - zwei Straßen S1, S2 zwischen Städten A und B
  - auf Straße S1 brauchen wir immer 1 Stunden von A nach B
  - auf Straße S2 hängt die Dauer von der Anzahl der Leute ab, die ebenfalls hier entlang fahren
  - Annahme 1000 Leute wollen von A nach B
  - möchte eigentlich das “Gemeinwohl” maximieren, d.h. die durchschnittliche Dauer der Fahrt von A nach B

“Price of Anarchy” Nash-Equilibria

- ein Nash-Equilibrium bezeichnet ein strategisches Gleichgewicht nicht-kooperativer Spieler, bei dem kein Spieler einen Anreiz hat, von seiner Strategie abzuweichen (eigentlich Spieltheorie)
- “Price of Anarchy”=Verhältnis des Gemeinwohls in einem Nash-Equilibrium zur optimalen (zentral geplanten) Strategie
- ja, das ist der Nash aus dem Kinofilm...
- Nash bekam 1994 den Nobelpreis für Wirtschaftswissenschaften

Braess-Paradoxon

- Dezentrale, “selfish” Routingschemata offenbaren noch andere, etwas seltsame Eigenschaften
- das Braess-Paradoxon zeigt, dass manchmal die Hinzufügung einer Handlungsalternative (hier: Bau einer Autobahn) die Gesamtlage verschlimmert (bei “selfish” Agents)

Braess-Paradoxon

- Beispiel:
  - 4 Städte S,A,B,T
  - 1000 Autos
  - Latenzfunktionen: 1 Stunde bzw. x/1000 Stunden und x/1000 Stunden bzw. 1 Stunde
  - Nash-Equilibrium: 1 ½ Stunden für alle auch optimal
  - Hinzufügung einer sehr schnellen Autobahn von B nach A verschlechtert das Nash-Equilibrium auf 2 !!

Mechanism Design / Pricing

- eine Sparte in der Mathematik/Informatik/ Wirtschaftswissenschaften befaßt sich damit, Kostenfunktionen so zu entwerfen, dass die Nash-Equilibria zu maximalem Gemeinwohl führen
- sehr aktives Forschungsgebiet

(De)zentrales Routing

- Aufgrund seiner Größe wird es immer schwieriger, den Datenverkehr des Internets zentral zu lenken
- Neuere Ansätze (insbesondere aus dem Peer-To-Peer Bereich) gehen eher in Richtung "selfish routing", d.h. jeder Sender/jedes Paket sucht sich selber die beste Route

Das Internet und sein Einfluß auf die Gesellschaft

- hatte das Internet zu Beginn eher Nischencharakter, ersetzt es immer mehr konventionelle Medien (Zeitungen, Fernsehen, Radio, Telefon, Post)
- viele “Internet-Firmen”
- es vollzieht sich auch ein Wandel der Nutzung vom passiven “Websurfer” zum aktiven Gestalter – Stichwort Web 2.0, Online Communities, Soziale Netzwerke
- überarbeitete Gesetze und Rechtsprechung notwendig, um den veränderten Gegebenheiten Rechnung zu tragen (Privatsphäre, SPAM, ...)

Das Internet als Informationsmedium

- Wie haben Sie sich bislang (vor dem Internet) über Dinge informiert, die Sie interessieren ?
- Bibliothek ...
- Zeitungen, Zeitschriften ...
- Fernsehen/Radio ...

Das Internet als Informationsmedium

- heutzutage “googlet” man oft nach diesen Informationen
- ... und nimmt dabei insgeheim an, dass “Google” (oder ein anderer Suchmaschinenbetreiber) eine ähnliche redaktionelle Auswahl vornimmt wie Bibliotheken, Zeitungen, Fernsehen/Radio
- stimmt das denn?

Auf der Suche nach Informationen im Internet

- zu Beginn des Internets waren “Internet Directories” sehr verbreitet (z.B. dir.yahoo.com)
- darin wurden (nach Ansicht der entsprechenden Redaktion) interessante Webseiten aufgelistet und kategorisiert
- durch den rapiden Zuwachs an Content im Internet ist es jedoch fast unmöglich ein derartiges Verzeichnis zu pflegen
- siehe später auch “Social Bookmarking”

Auf der Suche nach Informationen im Internet

- zwischenzeitlich haben sich Suchmaschinen auf breiter Front durchgesetzt
- zu Beginn: altavista, yahoo, ...
- heutzutage dominierend: Google
- Suchmaschinen durchforsten automatisiert das Internet, erfassen Webseiten und integrieren sie in einen Index, der typischerweise über ein Webinterface abgefragt werden kann

Suchmaschinen im Internet

- wenn Sie einen Suchbegriff - z.B. “LaTeX” - eingeben, liefert Google (oder eine andere moderne Suchmaschine) meistens unter den ersten “Hits” eine Seite mit den Informationen, die Sie gesucht haben
- wie macht das Google?
- es gibt sicherlich tausende oder gar Millionen von Seiten, auf denen das Wort “latex” – viele von denen enthalten sicherlich nicht die Informationen, die Sie suchen (einige davon wenden sich höchstwahrscheinlich auch eher an den begeisterten Gummiliebhaber)

Suchmaschinen im Internet

- Googles Suchmaschine besteht aus folgenden Komponenten
  - Web Crawler
  - Indexer
  - Pagerank
  - Searcher

Suchmaschinen im Internet

- Google verwaltet einen Index, in dem alle von Google registrierten Seiten, welche das Wort “latex” enthalten, aufgelistet sind
- wenn Sie nun “LaTeX” als Suchwort eingeben, wird Google in diese Liste schauen
- aber welche Webseiten stehen in dieser Liste oben?

Google PageRank

- der Google PageRank ist ein Verfahren, welches die “Wichtigkeit” einer Webseite schätzt
- der PageRank ist der Grund für die Dominanz von Google, sie waren die ersten, die ein solches Verfahren auf breiter Front eingesetzt haben
- aus der Liste der Webseiten die das Wort “latex” enthalten, werden die mit hohem PageRank zuerst ausgegeben

Googles PageRank

- der PageRank R(u) einer Webseite u ist ein Wert zwischen 0 und 1 (0...nicht wichtig, 1...sehr wichtig)
- der PageRank R(u) ist umso höher, je mehr Webseiten mit hohem Gewicht auf u verweisen

$$PR_i = \frac{1-d}{N} + d \sum_{\forall j \in \{(j,i)\}} \frac{PR_j}{C_j}$$

- hierbei ist N die Anzahl der erfaßten Webseiten und d ein Dämpfungsparameter; Cj ist die Anzahl der Webseiten, auf die Cj verweist
- Idee: jede Seite “verteilt” seinen Pagerank auf seine Links

Googles PageRank

- das sieht wie ein Henne-Ei Problem aus
- wie kann ich den PageRank R(u) berechnen, wenn ich dazu die PageRanks aller Seiten kennen muss, die auf u verweisen?

Googles PageRank

- Man kann den PageRank R(u) einer Seite u interpretieren als die Wahrscheinlichkeit, mit der ein “zufälliger Surfer” auf Seite u landet
- der “zufällige Surfer” wählt mit Wahrscheinlichkeit 1-d eine beliebige neue Seite, mit Wahrscheinlichkeit d besucht er eine zufällige der in der aktuellen Seite verlinkten Seiten
- auf “wichtigen” Seiten hält sich der Surfer auf lange Sicht öfter auf

Googles PageRank

- den PageRank R(u) könnte man daher auch dadurch ermitteln, dass man lange genug “zufällig” im Netz surft, und beobachtet, wie oft man welche Seite besucht
- mathematisch läßt sich die Berechnung des PageRanks aber auch als Eigenvektorproblem darstellen und lösen

Probleme des PageRanks

- Ranking hat nichts mit der Qualität des Inhaltes einer Seite zu tun
- Ranking basiert nur darauf, wie die Webseite mit anderen Webseiten verlinkt sind
- man kann versuchen, den PageRank bewußt zu manipulieren, z.B. durch Linknetzwerke

Social Bookmarking

- gemeinschaftlich erzeugte Menge von “Lesezeichen”
- Beispiele: del.icio.us, digg, Mister Wong, yigg ...
- Nutzer können Lesezeichen hinzufügen, löschen, kommentieren und “taggen”
- Lesezeichen können mit anderen Nutzern geteilt werden