Hilfe, ich bin im Internet!

Doris Diedrich

Wie schütze ich mich gegen Hacker, Viren, Trojaner und andere Frechdachse?

Übersicht

- Was ist das Problem?
- Wie funktioniert das Internet?
- Wo sind Angriffspunkte?
- Was kann ich dagegen tun?

Was ist das Problem?

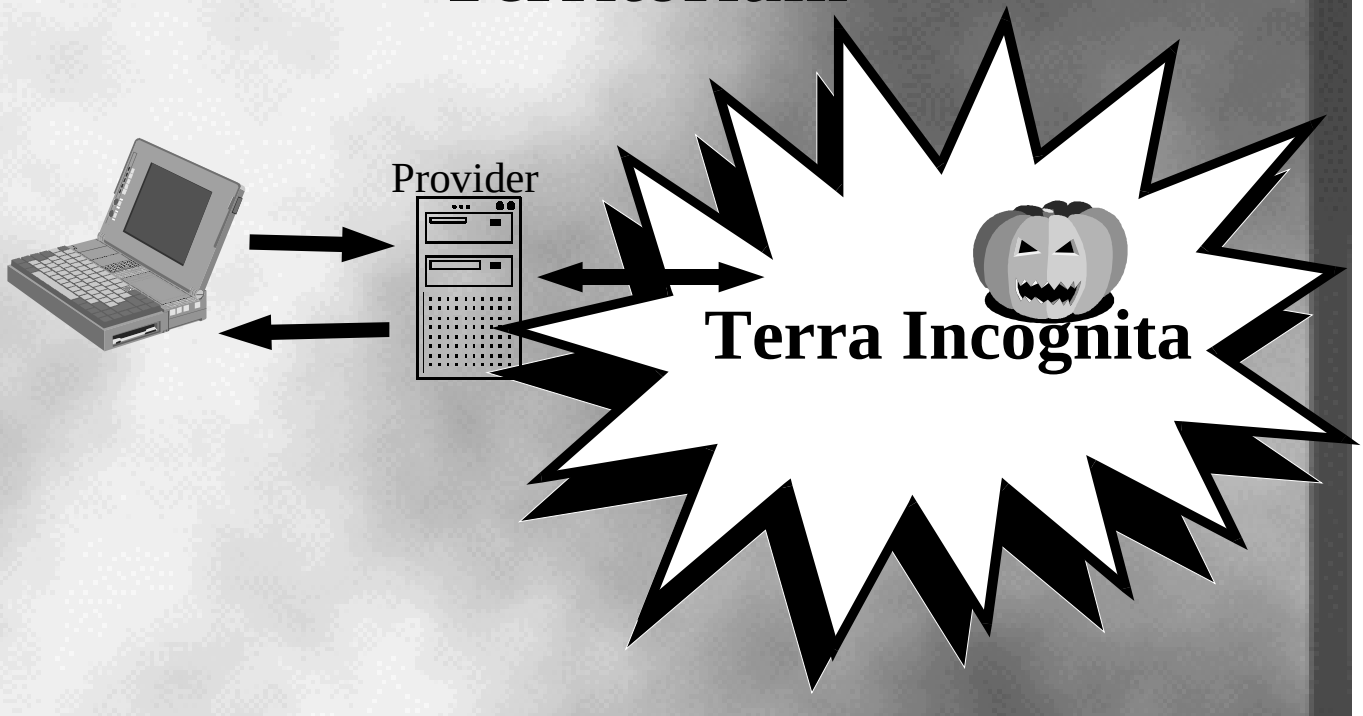
- Keine Privatsphäre
- Finanzieller Schaden
- Schaden am PC durch Viren
- Der eigene PC als Plattform für Angriffe
- Langsame Internetverbindung
- ...

Angriffe

- Abhören von Verbindungen (Emails, Passwörtern, Daten, die ich an meinen Online-Buchhändler schicke...)
- Virenbaukasten
- Angriffe durch fertige Tools (Scriptkiddies)
- Für Alles gibt's Tools...
- Beispiel: das vergleichsweise komplizierte Ethereal zum Abhören von Verbindungen

Ethereal Demo

Das Internet: unbekanntes Territorium



Motto

Ein System ist nur so sicher, wie das unsicherste seiner Elemente!

Es gibt keine Sicherheit, es gibt nur verschiedene Grade der Unsicherheit!

Das bedeutet: man kann seinen Rechner *ziemlich* sicher kriegen. Absolute Sicherheit gibt es auch: man ziehe den Stecker!

Wie funktioniert das Internet Schritt für Schritt?

- Kontaktaufnahme mit dem Provider: per Modem einwählen.
- Die Kommunikation PC <->Provider steht, weiter geht's ins Internet!
- Ich will eine http Seite: Nachricht an den passenden Server schicken
- Seite anschauen!

Einschub: Client Server

- Server stellen eine Dienstleistung bereit. Es sind Programme und Computer, die i.A. dauerhaft ans Internet angeschlossen sind und darauf warten, von Clients angesprochen zu werden
- Clients sind Programme/Computer, die Dienstleitungen von den Servern anfordern

Protokolle

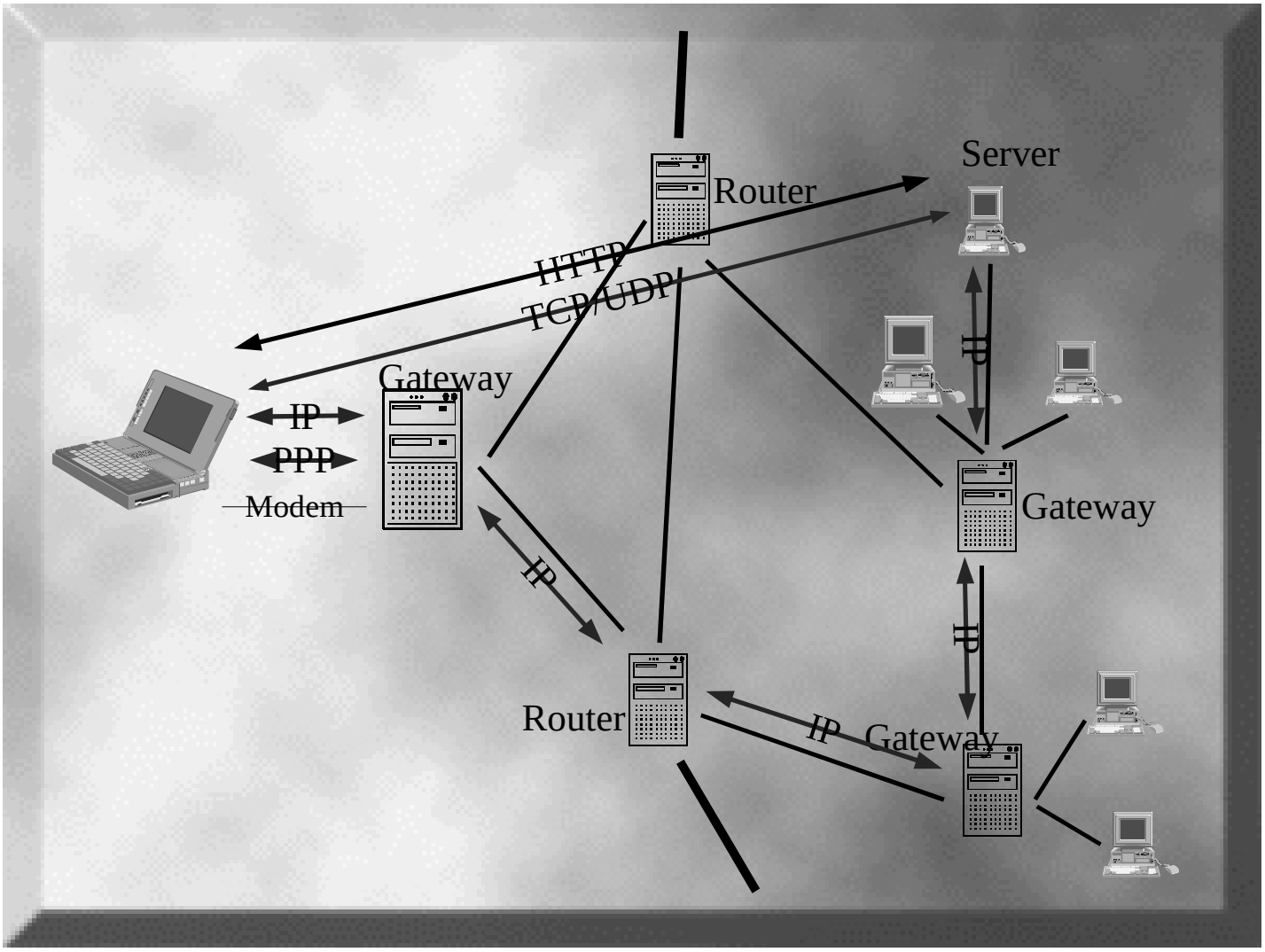
Damit Kommunikation stattfinden kann, braucht es Regeln und festgelegte Abläufe.

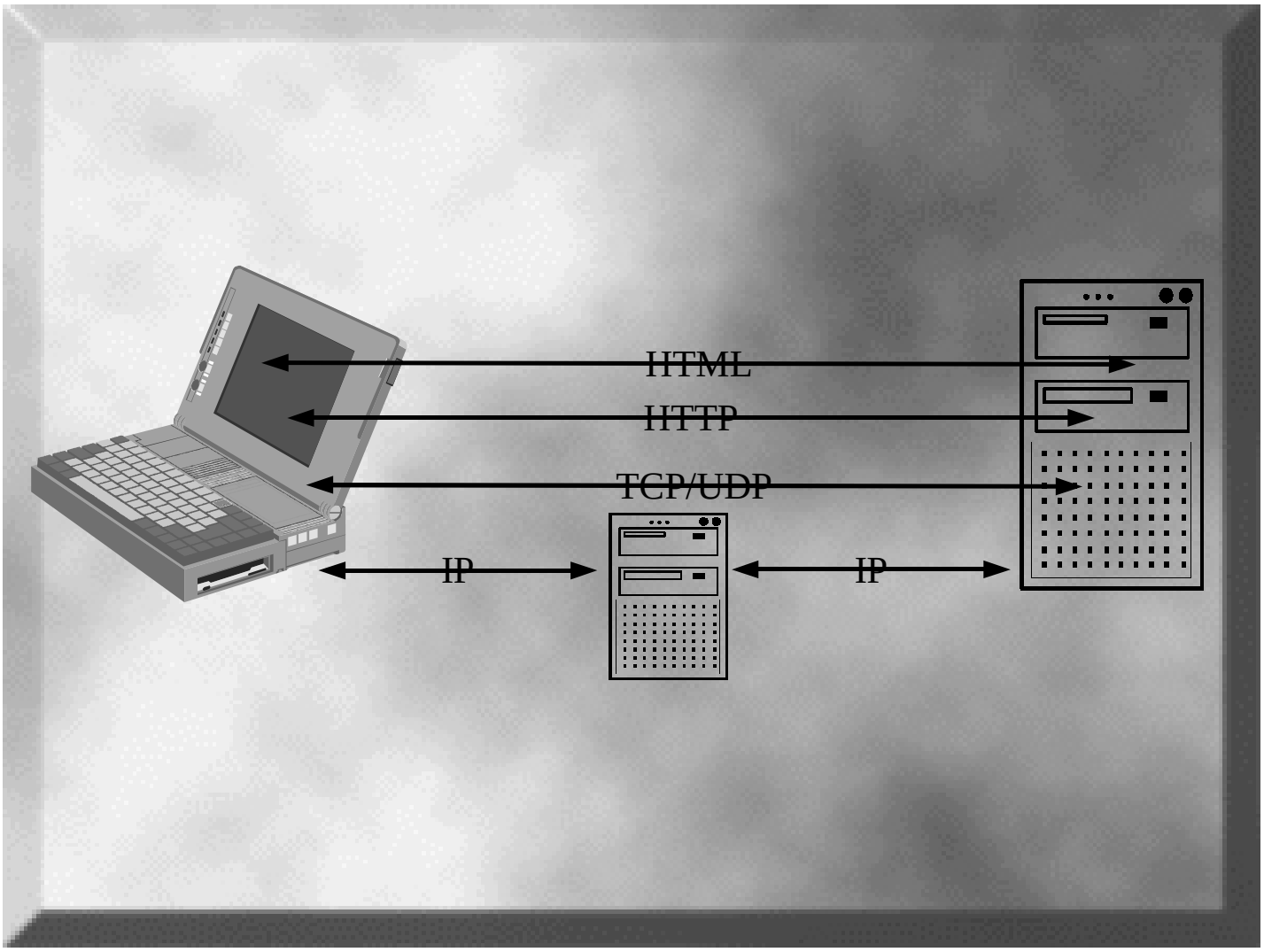
- Guten Tag!
- Ebenfalls guten Tag!
- Wissen Sie, wie spät es ist?
- Ja!

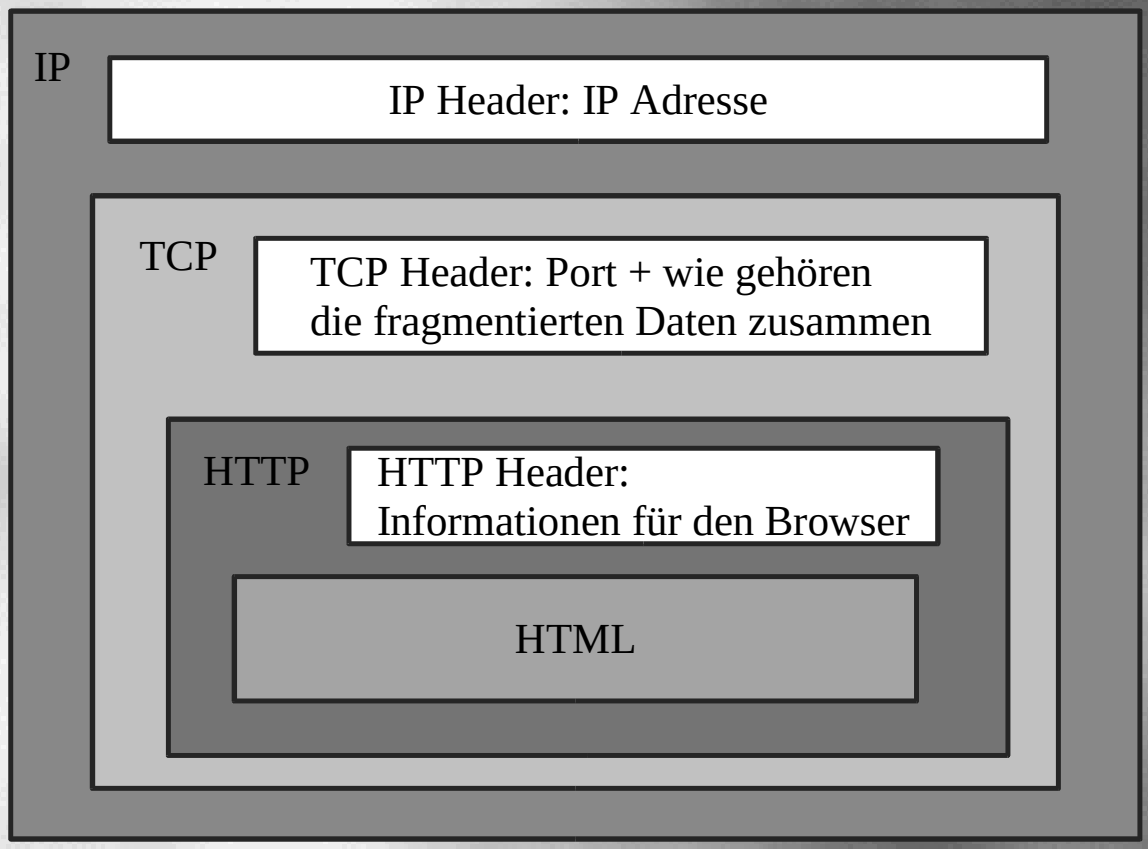
-> Ein Protokoll!

- Verschiedene Protokolle für verschiedene Zwecke:

jeder Zweck verlangt andere Abläufe!


Server
Router
HTTP
TCP/UDP
IP
Gateway
IP
PPP
Modem
IP
Gateway
IP
Router
IP
Gateway


HTML
HTTP
TCP/UDP
IP
IP



PPP Point to Point Protokoll

- Zur Kommunikation zwischen zwei Punkten
- Oft für Modemverbindungen
- Wird oft verwendet für die Verbindung: PC -> Gateway des Providers ins Internet
- PPP sorgt dafür, dass meinem Rechner eine IP Adresse zugeteilt wird

IP Internet Protokoll

- legt fest, wie Internet Adressen aussehen, wie sie verteilt sind (ZB 217.49.26.115)
- Netze haben Nummern, deren Subnetze haben ebenfalls Nummern, deren Subnetze wiederrum Insgesamt gibt es 4 Hierarchiestufen
- Aus der IP-Adresse wird festgestellt, in welches Netz und an welchen nächsten Router ein Paket weiter geleitet werden soll
- Jeder Rechner hat eine IP Adresse
- Mein PC kriegt die IP Adresse vom Provider zugeteilt, nachdem er diesen angerufen und um eine Internetverbindung gebeten hat

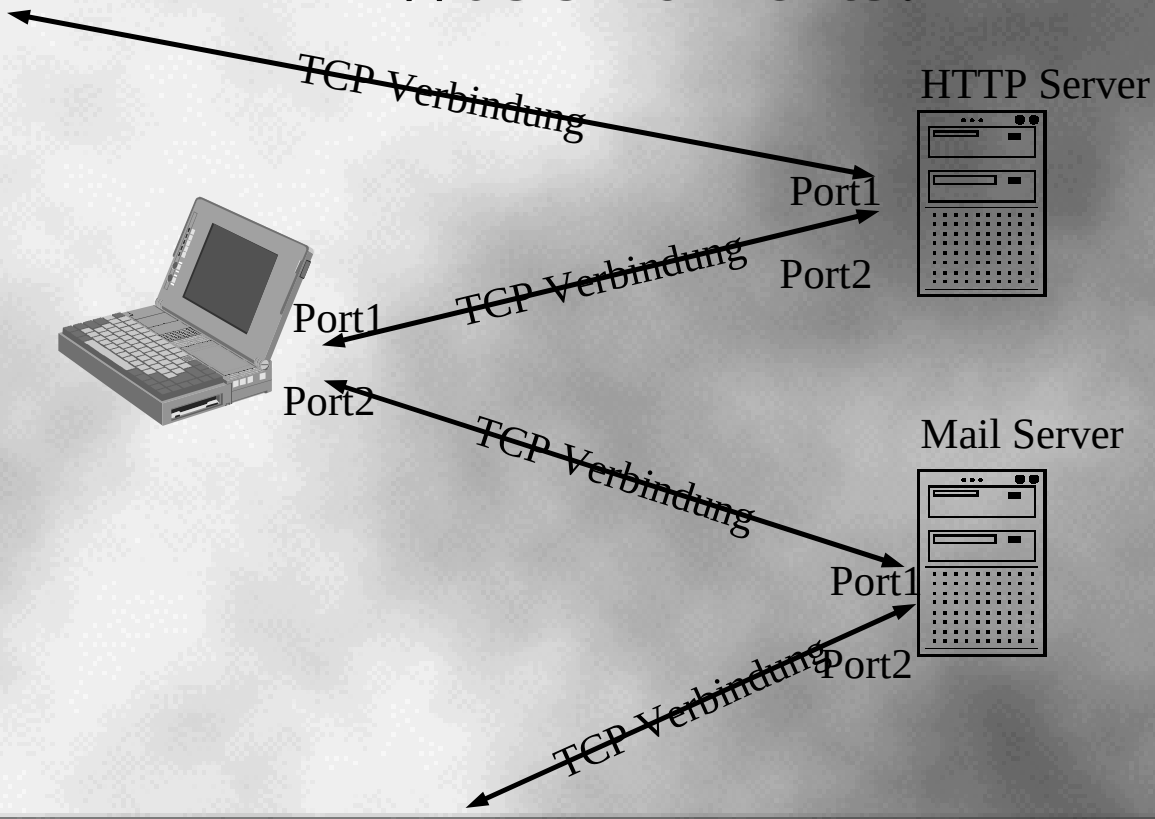
DNS

- Domain Name Service
- Übersetzt Adressen in Klartext in Internet Adressen
- Diese IP-Adressen, zB 217.89.94.22 werden von den Routern angeschaut, um die Pakete weiter leiten zu können
- Eigene DNS Server, hierarchisch im Internet verteilt, im Prinzip hat jedes Netz mindestens einen DNS Server
- Die IP-Adresse des DNS Servers muss in der TCP/IP Konfiguration angegeben werden oder wird automatisch vom Provider zugeteilt (PPP)

TCP: verlässlicher Transport

- TCP Transport Control Protocol
- TCP setzt auf IP auf und tut so, als würde es direkt von Programm zu Programm kommunizieren.
- TCP fragmentiert zu schickende Daten (zB eine Webseite) in kleine transportgrosse Stücke
- TCP sorgt dafür, dass die Pakete am Ziel in richtiger Reihenfolge und korrekt ankommen
- TCP fordert verloren gegangene Pakete erneut an
- TCP beobachtet die Frequenz mit der die Pakete ankommen und bei wenig Durchsatz schickt es auch weniger Pakete los (flow control)
- TCP macht „congestion control“, „Grid locks „ verhindern

Was sind Ports?



Ports

- Server „lauschen“ an vorher festgelegten Ports.
- Wenn ich einen Webserver kontaktiere, dann weiss ich, dass ich an die Adresse des Servers und an den Port 80 schicken muss. Dieser Port ist i.A. für Webserver belegt.
- Im Prinzip kann ein Programm aber an jedem Port lauschen: nach der Kontaktaufnahme findet die eigentliche Kommunikation aber über einen neuen zufälligen Port statt.

HTTP

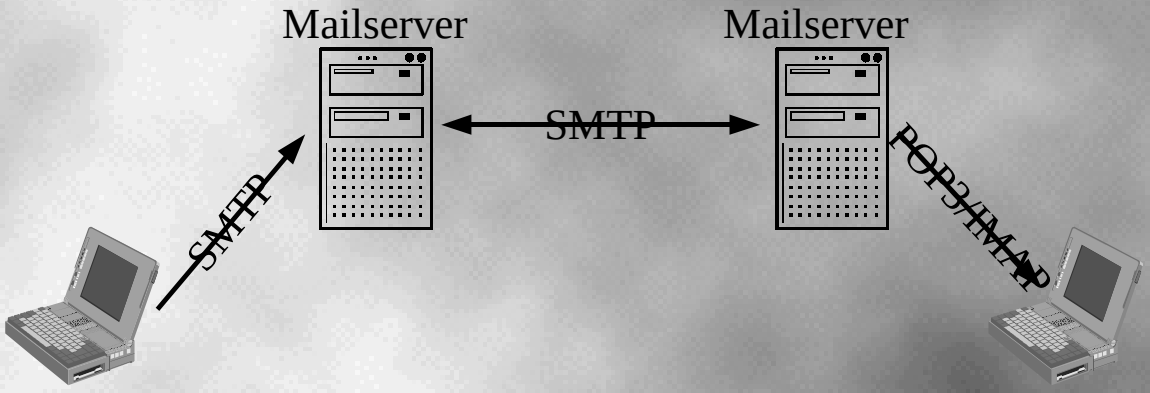
- HTTP HyperText Transport Protocoll
- Http ist speziell auf den Transport von HTML Seiten zugeschnitten
- Es sorgt zB dafür, dass unveränderte Seiten, die bereits im Browser Cache liegen, nicht noch einmal neu geschickt werden
- Es sorgt auch dafür, dass der Server bestimmte Dinge über den Client erfährt (Browsertyp, Sprache (dt,fr...) u.a.), dass Cookies geschickt, Passwörter angefordert werden, es schickt die URL an den Server...

Beispiel HTTP Protokoll

- GET /somedir/page.html HTTP/1.1
  Connection: close
  User-agent: Mozilla/4.0
  Accept: text/html, image/gif, image/jpeg
  Accept-language:fr
  Cookie: DorisDiedrich
  If-modified-since: Mon 20 Jun 2001 09:23:24
- HTTP/1.1 200 OK
  Connection: close
  Date Thu, 06 Aug 2003 12:00:15 GMT
  Server: Apache/1.3.0 (Unix)
  Last-Modified: Mon, 22 Jun 2002 09:23:24 GMT
  Set-cookie: DorisDiedrich
  Content-Length: 6821
  Content-Type: Text/html
  data data data

SMTP

- simple Mail Transfer Protokoll
- Wird genutzt, um Emails an einen Mailserver zu schicken und von einem Emailserver zum anderen zu transportieren



SMTP Beispiel

S: 220 hamburger.edu
C: HELO crepes.fr
S: 250 Hello crepes.fr, pleased to meet you
C: MAIL FROM:<alice@crepes.fr>
S: 250 alice@crepes.fr...Sender ok
C: RCPT TO: <bob@hamburger.edu>
S: 250 bob@hamburger.edu...Recipient ok
C: DATA
S: 354 Enter mail, end with „.“ on a line by itself
C: Hallo, wann soll es dieses Jahr schneien?
C: Es gruessen die Schneemaenner.
C: .
S: 250 Message accepted for delivery
C: QUIT
S: 221 hamburger.edu closing connection

Mailformat

From: alice@crepes.fr
To bob@hamburger.edu
Subject: Was ist der Sinn des Lebens?
AndereHeaderTeile: xyz
MIME-Version: 1.0
Content-Type:multipart/mixed;Boundary=StartOfNextPart

--StartOfNextPart
Und, hast Du den Sinn gefunden? Vielleicht hilft das Bild!
--StartOfNextPart
Content-Transfer-Encoding: base64
Content-Type:image/jpeg
base64 encoded data ....
.....base64 encoded data

POP3- Post Office Protocol Version3

- Nur der Email Server ist konstant mit dem Internet verbunden und kann Email in Empfang nehmen, wenn sie gerade eintrifft: smtp
- PCs holen die Email vom Server, wenn sie gerade mal am Netz sind: POP3/IMAP
  - POP3 holt Email vom Server ab
  - Authentifiziert den Benutzer beim Server (Passwort und login)
  - „download and keep“ oder „download and delete“

IMAP - Internet Mail Acces Protocol

- IMAP holt Email beim Server ab
- deutlich mehr Optionen als POP3
- Erlaubt auch, nur Teile (zB nur das Subject) von Emails downzuloaden
- Erlaubt so die Auswahl von Nachrichten zum downloaden
- Andere Features (Suchen in den Nachrichten auf dem Server, etc)
- Letztlich hängen die tatsächlichen Möglichkeiten vom Email-Client ab

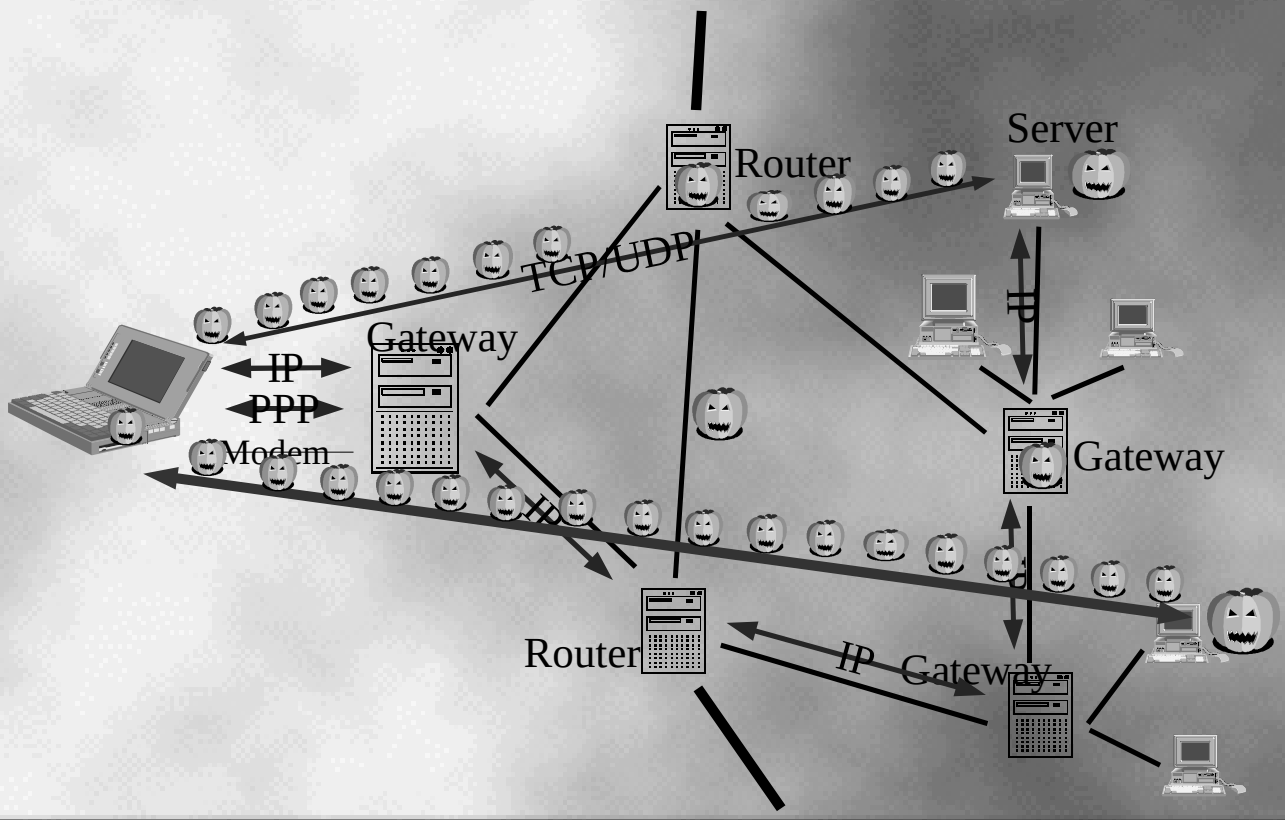
Viele weitere Protokolle

- Es werden viele weitere Protokolle genutzt um das Internet „am Laufen“ zu halten. ZB zum Austausch und zur Aktualisierung von Routingtabellen
- Fast jede Anwendung hat ein eigenes Protokoll auf Application Layer Ebene: ZB IRC, News, Email (SMTP und andere), HTTP...
- Jedes Transportmedium (zB Koaxkabel) hat mindestens ein Protokoll Transport der Bits
- ...

Internet Layer ISO/OSI

- Application Layer: Transport ganzer Dateien, zB HTTP, SMTP, IMAP, andere...
- Transport Layer: Transport von Paketen TCP oder UDP
- Network Layer: Routet (leitet) TCP/UDP-Pakete von einem Host zum nächsten. IP und Routing Protokolle
- Link Layer: Transport von Paketen auf einer Leitung: Ethernet, PPP...
- Physical Layer: Transport der einzelnen Bits auf Leitungen. Hängt vom Medium ab.

Gefahren im Internet



Gefahren im Internet

- Denial of Service
- Abhören von Leitungen
- Verbreitung von Viren, Trojanern, Würmern über Programme, Dateien, Webseiten...
- Einschleusen von Dialern über Webseiten und Programme
- Hacker, die Sicherheitslücken ausnutzen mit Portscans
- Fälschen von Absendern von Emails, Webseiten, etc
- Falsche Webseiten durch DNS Angriffe, Überlisten der Benutzer...
- ... täglich neue Methoden...

Sicherung

- 2 teiliges Konzept zur Gefahrenabwehr
  - Gefahrenabwehr am PC (bzw am Netz) direkt
  - Gefahrenabwehr durch sichern der Verbindungen nach draussen: SSL, SSH, Verschlüsselung

Wie sichert man Verbindungen nach Draussen? Da Verschlüsselung eine Grundlage der anderen Techniken ist, fangen wir damit mal an...

Gliederung Verschlüsselung/ Einführung Kryptographie

- Allgemeine Fakten
- Symmetrische Verschlüsselung
- Asymmetrische Verschlüsselung
- Mythos „Beweisbar sicher“
- Schlüsselaustausch: PKI und Signaturen
- SSH: ein einfaches Verfahren?
- SSL: symmetrische und asymmetrische Verfahren kombiniert, Zertifikate

Verschlüsselung: Einige Fakten

- Der Algorithmus mit dem verschlüsselt- bzw entschlüsselt wird, ist allgemein bekannt (Kerkhoffsches Prinzip)
- Algorithmen werden durch Kryptanalytiker getestet->wenn der Algorithmus gebrochen wird, wird das frühzeitig bekannt
- Ein Algorithmus ist nur so gut, wie seine Implementierung!
(The generation of random numbers is too important to be left to chance.)

Grad der Sicherheit

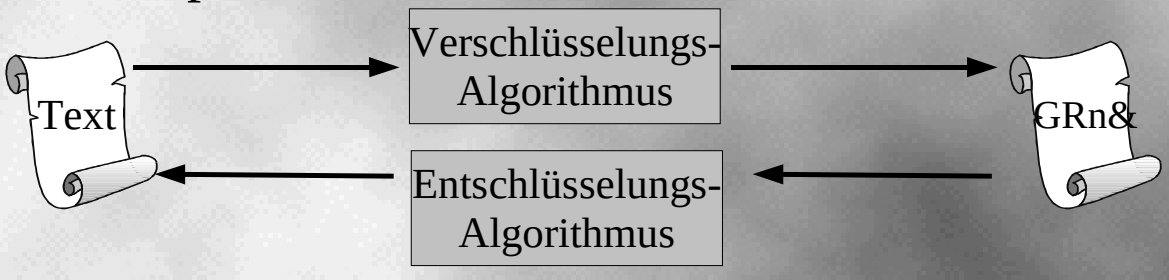
- Informationstheoretische Sicherheit:
Absolut sicheres Verschlüsselungssystem: kann nicht geknackt werden (bei korrekter Verwendung)
- Kryptographische Sicherheit:
es ist komplexitätstheoretisch schwer, den Schlüssel zu finden bzw eine korrekte Entschlüsselung hinzukriegen

Sicherheitsziele

- Geheim
- Authentisch: wirklich vom angegebenen Absender
- Integer: unverändert
- Anonym: Absender/Adressat geheim

Symmetrische Verschlüsselung

- Ziel Geheimhaltung
- Jede Partei besitzt denselben Schlüssel
- Der Schlüssel muss geheim gehalten werden!
- Alice verschlüsselt mit demselben Schlüssel, mit dem Bob entschlüsselt!
- Beispiele: 3DES, OneTimePad...



Beispiel für Symmetrische Verschlüsselung: One-Time-Pad

- Schlüssel: Zufällige Bitfolge
  01000100111100
- Verfahren: Zu verschlüsselnde Datei wird in Bitfolge übersetzt, der Schlüssel wird binär aufaddiert
- Verschlüsselung: Entschlüsselung:

Schlüssel	01000100111100 +	01000100111100 +
	01111010000010	00111110111110
	---	---
	00111110111110	01111010000010

Probleme

- Problem: ist die Bitfolge (Schlüssel) vorhersagbar (d.h. nicht zufällig) dann ist das Verfahren sinnlos (ausprobieren!)
- Kryptanalyse One-Time-Pad: versuchen Sie, Schlüssel mehrfach zu verwenden: ein Kryptanalytiker kommt zum Ziel, indem er Schlüsseltexte aufaddiert (ausprobieren!)
- Deshalb bei One-Time-Pad: Der Schlüssel ist mindestens so lang wie die zu verschlüsselnde Datei, und nur einmal zu verwenden! Dann: Absolute Sicherheit!
- Problem: Schlüsselaustausch

Symmetrische Verschlüsselung 2

- Häufiger Schlüsselaustausch nicht überall möglich oder praktisch
- Systeme, deren Schlüssel mehrfach verwendbar sind
  - Problem hier: oft Kryptanalyse möglich
  - Brute-Force Angriff möglich: alle in Frage kommenden Schlüssel ausprobieren (das Verschlüsselungsverfahren ist ja allgemein bekannt)
  - Keine Informationstheoretische Sicherheit mehr

Mythos: „Beweisbar Sicher“

- Beweisbar sichere Systeme: HURRAH!!!! Keine Kryptanalyse?
- Einzige Systeme, die bisher beweisbar sicher sind: One-Time-Pad und El Gamal
- Der Beweis für El-Gamal wird folgendermassen geführt:
Man beweist, dass das Verfahren sicher ist, unter der Voraussetzung, dass die Annahme stimmt, die dem Verfahren zugrunde liegt. Die Annahme zu beweisen ist aber NP-Vollständig.
- Komplexitätstheoretisch sicher: es ist SCHWER, den richtigen Schlüssel/Klartext zu finden.Je länger der Schlüssel, desto schwerer ist

Andere Lösungen

- Komplexitätstheoretisch sicher sei uns genug. Informationstheoretisch (absolut) sicher ist nur One-Time-Pad=> El Gamal, RSA, IDEA, 3DES etc genügen uns erstmal...
- Schlüsselaustausch bleibt ein Problem auch bei mehrfach verwendbaren Schlüsseln. Manchmal (Web) kommunizieren Parteien, die sich gar nicht kennen
- Neue Verfahren zum Schlüsselaustausch (Diffie-Hellmann)

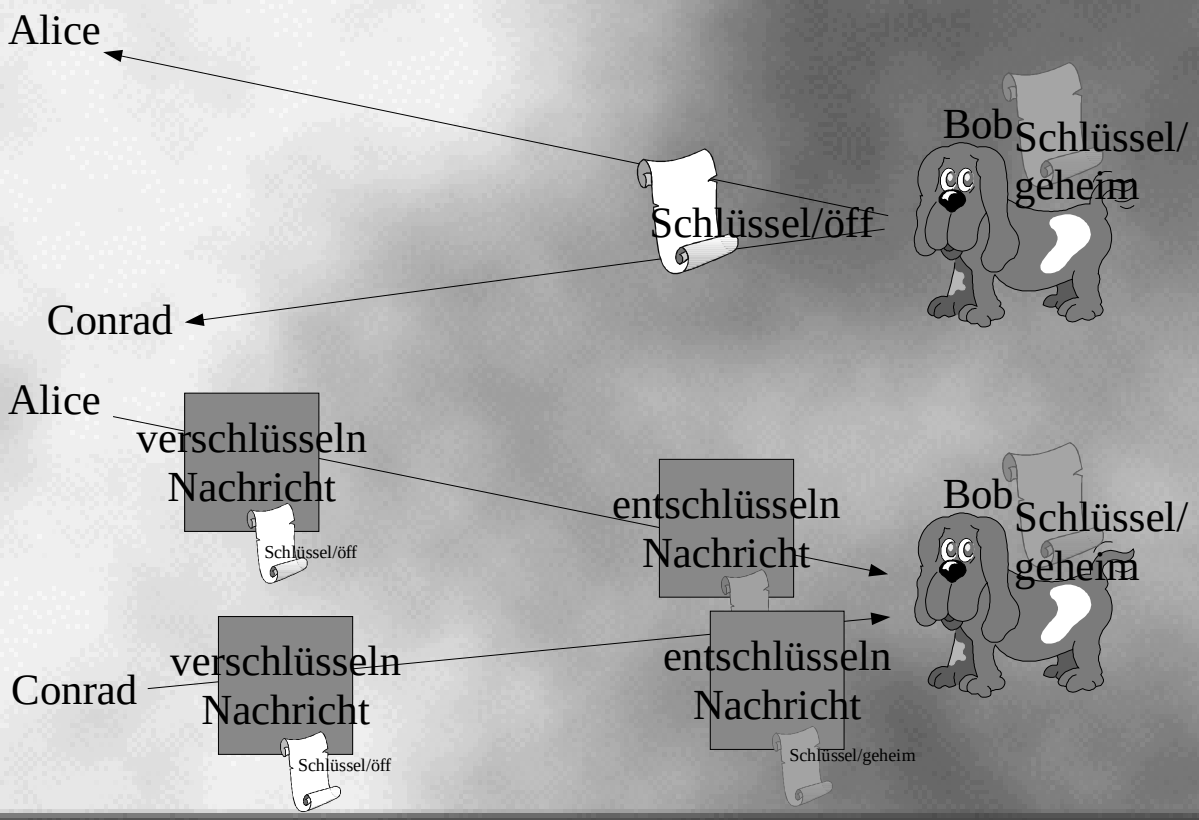
Diffie-Hellmann Schlüsselaustausch

- Beruht auf der Annahme, dass es schwer ist, den diskreten Log in endlichen Körpern zu ziehen
- Alice und Bob einigen sich auf n (prim) und g
- Alice schickt Bob: $g^x \bmod n$
- Bob schickt Alice: $g^y \bmod n$
- Beide berechnen den geheimen symmetrischen Schlüssel g^{xy}

Asymmetrische Kryptographie

- Verfahren, bei denen unterschiedliche Schlüssel zum Verschlüsseln und Entschlüsseln gebraucht werden=> öffentliche Schlüssel!

Asymmetrische Verschlüsselung



Asymmetrische Kryptographie

Ich möchte Bob etwas schicken, dass nur er lesen können soll.

- Ich bitte Conrad, mir Bobs öffentlichen Schlüssel zu geben (den darf jeder kennen)
- Es ist schwer, aus dem öffentlichen Schlüssel, Bobs geheimen Schlüssel zu berechnen, deshalb versuch ich es gar nicht erst
- Ich verschlüssele mit Bobs öffentlichem Schlüssel und schicke Bob die verschlüsselten Daten über einen unsicheren (öffentlichen) Kanal.
- Bob entschlüsselt mit seinem geheimen Schlüssel.

RSA

- Ein klassisches asymmetrisches Verschlüsselungsverfahren ist RSA (Rivest, Shamir, Adleman)
- Annahme: es ist schwer, große Zahlen in ihre Primfaktoren zu zerlegen
- (Unbewiesene) Annahme: nur wenn man eine Zahl in ihre Primfaktoren zerlegen kann, dann kann man RSA brechen
- $n = p * q$ p,q Primzahlen, e zufällig und relativ prim zu $(p-1)(q-1)$
- Berechne $d = e^{-1} \bmod ((p-1)(q-1))$

Fortsetzung RSA

- Blockweise Verschlüsseln:
  $c = m^e \bmod n$ m Text, c Schlüsseltext
- Blockweise Entschlüsseln:
  $m = c^d \bmod n$ $d = e^{-1} \bmod ((p-1)(q-1))$
- Schlüssel zum Verschlüsseln:
  n, e
- Schlüssel zum Entschlüsseln:
  n, d

Beispiel RSA

- p=47 q=71 Primzahlen
- n=p*q=3337
- e zufällig 79
- (p-1)(q-1)=46*70=3220
- d=79-1 mod 3220 = 1019
- Sei m= 6882326879666683
- m_1 = 688 m_2=232 ...
- $688^{79} \bmod 3337 = 1570 = c_1$
- $1570^{1019} \bmod 3337 = 688 = m_1$

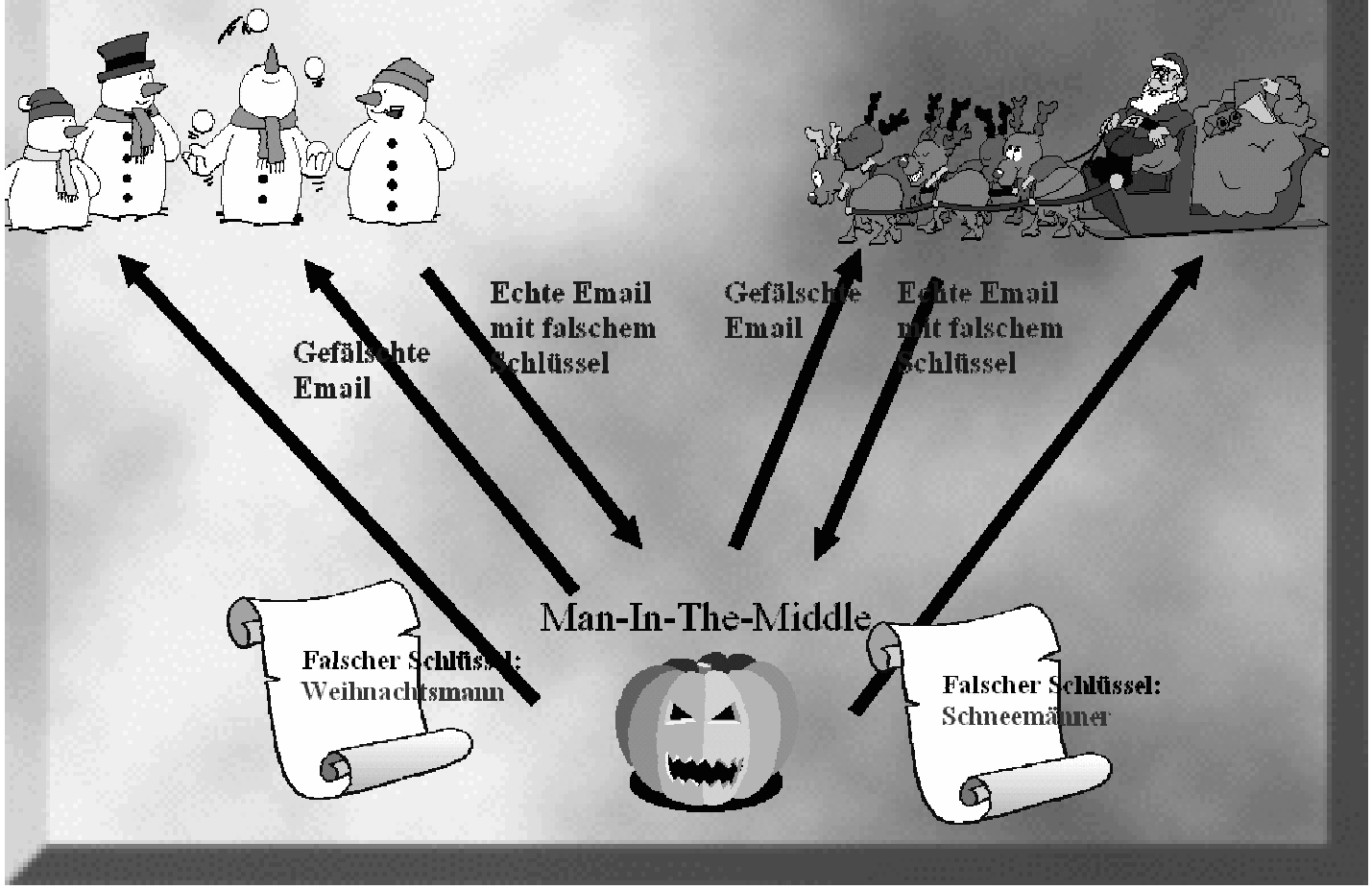
Verschlüsselte Email

- Eine Anwendung der asymmetrischen Kryptographie: verschlüsselte Email
- Alice will mir eine Email schicken, aber sie will nicht, dass Jede sie lesen kann
- Alice verschlüsselt mit meinem öffentlichen Schlüssel und ich entschlüssele mit dem einzig mir bekannten einzig passenden geheimen Schlüssel.

Probleme

- Woher weiss ich, dass ich den richtigen Schlüssel habe?
- Mallory könnte mir seinen eigenen Schlüssel unterjubeln, nach dem Lesen mit dem richtigen Schlüssel verschlüsseln und die Email an Bob schicken.
- Falls Alice und Bob nicht mal zufällig zusammen kommen und Schlüssel vergleichen, krachts. Und Keine wird's je merken...

Man In The Middle



Der Man-In-The-Middle Angriff

- Nehmen wir an, die Schneemänner möchten vom Weihnachtsmann wissen, wann sie es schneien lassen sollen
- Der große Kürbis hat was dagegen und gibt den Schneemännern heimlich seinen eigenen Schlüssel
- Der große Kürbis fälscht eine Email und signiert sie mit dem untergejubelten Schlüssel
- Die Schneemänner testen die Signatur/Email und komme zu dem Schluss, dass sie vom Weihnachtsmann stammt.
- Wieder kein Schnee zu Weihnachten!

Lösung?

- Die öffentlichen Schlüssel zB des Weihnachtsmannes und der Schneemänner werden von Personen signiert, deren Schlüssel ich sicher kenne, die ich zB getroffen habe und Schlüssel direkt ausgetauscht habe
- Oft: die öffentlichen Schlüssel werden von öffentlichen Zertifizierungsstellen signiert (unter strengen Bedingungen).
  - Einer zentralen Instanz muss man absolut vertrauen können. Ihr Signierschlüssel muss 100% sicher sein, sonst sind alle Schlüssel unsicher.
  - Aber: wir sind alle nur Menschen...
- Schlüsselabgleich vereinfachen!

Schlüsselvergleich vereinfachen

- Ich möchte einen über unsichere Kanäle geschickten Schlüssel mit einem mir bekannten Schlüssel vergleichen, damit ich ihn signieren kann: ist der Schlüssel korrekt? (Vermeidung eines Man-In-The-Middle Angriffes) Nur dann will ich ihn nutzen/signieren
- Schlüssel sind lang
- Lösung: Es wird nur der „Fingerabdruck“ des Schlüssels verglichen (gleiches System wird auch bei Schlüsselaustausch zum Email sichern verwendet)

Schlüsselvergleich

- Ich lade mir Alice Schlüssel von ihrer Webseite herunter
- Da die Webseite gefälscht sein könnte, suche ich eine Email von Alice, in der sie mir den Fingerprint des Schlüssels geschickt hatte
- Dann berechne ich den Fingerprint des Schlüssels und vergleiche
- Zwei unsichere Kanäle: aber ein Angreifer hat es schwer, beide gleichzeitig zu blocken/anzugreifen

Hashfunktionen

- Hashfunktionen verkürzen den Schlüssel indem sie aus dem Schlüssel einen Wert mit kürzerer fester Länge berechnen (Fingerprint).
- Der Hinweg ist leicht, der Rückweg, also die Berechnung des zugehörigen Schlüssels, ist schwer (sonst kann ich anderen Schlüssel berechnen und unterjubeln, die zum Fingerprint passen)
- Hashfunktionen sollten möglichst kollisionsfrei (d.h in der Kryptographie: kollisionsarm) sein, damit kein anderer Schlüssel mit demselben Hashwert (Fingerprint) gefunden werden kann.

Vertrauen in Schlüssel

- Kann ich einem so gewonnenen Schlüssel vertrauen?
- Vielleicht trau ich mich noch, Alice jetzt meine Reisedaten verschlüsselt zu schicken.
- Aber wieweit kann ich so einem Schlüssel trauen?
- Besser: Ich rufe Alice an und lass mir von ihr den Fingerprint durchgeben, ich kenn ja ihre Stimme...

Signaturen

- Nicht alle Leute, denen ich etwas schicken möchte, kenne ich persönlich. Unnu?
- Signaturen zum Unterschreiben von Schlüsseln: „Die UNterzeichnende bestätigt, dass dieser Schlüssel echt der Person x gehört“

Signaturen

- Signaturen sind Elektronische Unterschriften: „Ich bestätige die Richtigkeit“
- Signaturen sind bei RSA ein „Nebenprodukt“ der asymmetrischen Verschlüsselung
- Ich kriege einen Schlüssel: wenn dieser den Schlüsseltext entschlüsselt, weiss ich, dass die Verschlüsselung NUR mit dem PASSENDEN eindeutigen Verschlüsselungsschlüssel erfolgt sein kann: die Datei ist also signiert worden.
- Wie kann ich sicher sein, dass der Schlüssel, den ich gekriegt habe, auch wirklich zu Bob gehört?

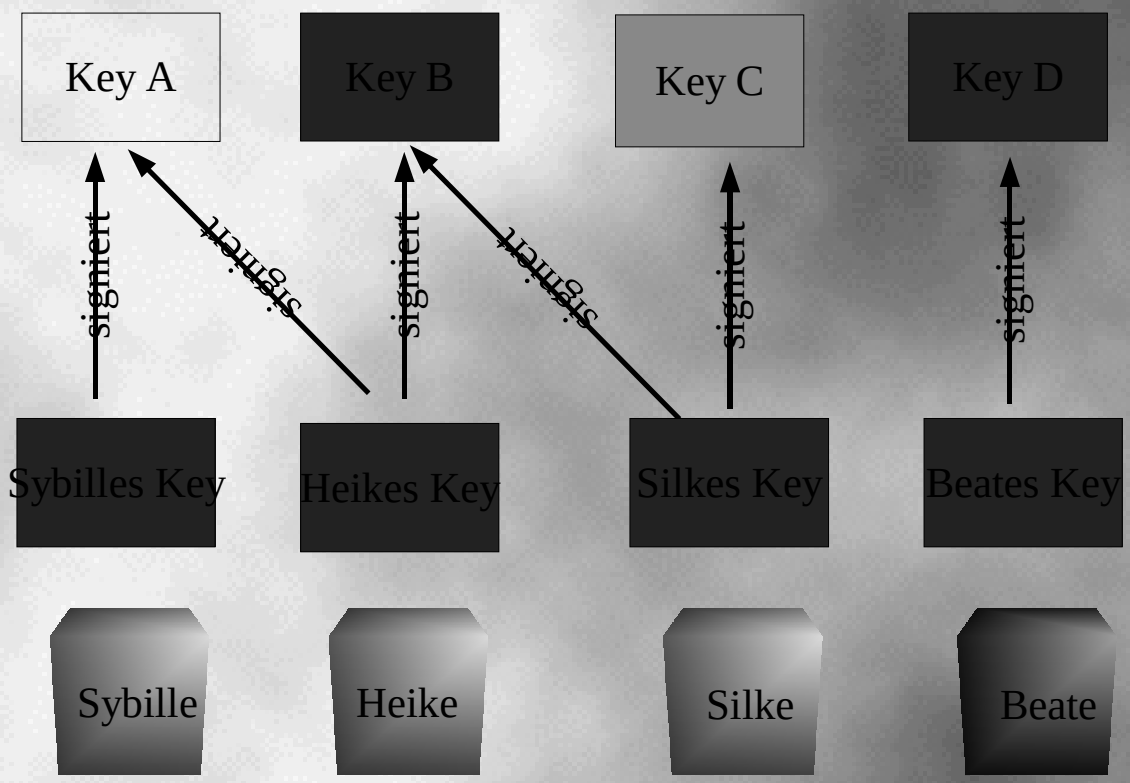
Signatur

- Alice verteilt ihren öffentlichen Schlüssel grosszügig.
- Alice schickt mir einen Schlüssel und hängt die Signatur an. Die Signatur ist einfach der Text, mit Alice's geheimem Schlüssel verschlüsselt
- Ich entschlüssele die Signatur mit dem mir bekannten öffentlichen Schlüssel
- Ich vergleiche die Schlüssel: stimmen sie überein?

Public Key Infrastructure Web-of-Trust

- PGP (GnuPG) bietet einen Mechanismus zur Schlüsselverteilung, der sich Web-Of-Trust nennt
- Jeder Benutzer generiert und verteilt seinen eigenen öffentlichen Schlüssel über Server. Die Benutzer unterzeichnen ihre Schlüssel gegenseitig und schaffen so ein Web-Of-Trust ohne zentrale Instanz zur Zertifizierung.
- Dabei können Sie Schlüsseln und Menschen unterschiedlich stark vertrauen:
  - Signiert/Prüft eine Person zuverlässig?
  - Stammt der Schlüssel aus sicherer Quelle?

Web of Trust



Demo GnuPG

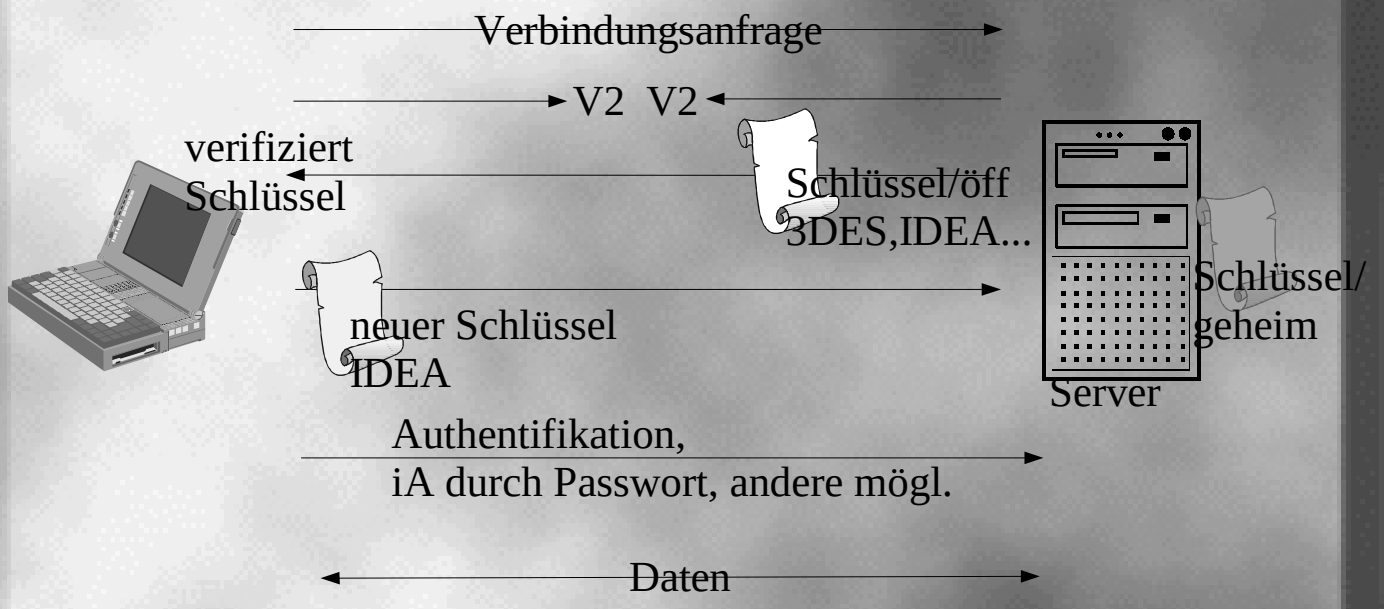
El Gamal

- Reines El Gamal zum Signieren, Variante zum Verschlüsseln
- Annahme auch hier: das Verfahren ist sicher, wenn die Annahme stimmt, dass es schwer ist, diskrete Logarithmen über einem endlichen Körper zu bestimmen
- Primzahl p Zufallszahlen g und $x < p$
- Berechne $y = g^x \bmod p$
- Öffentlicher Schlüssel y, g und p
- Privater Schlüssel x

Verbindungen sichern

- Verbindungen ins Internet werden durch Protokolle gesichert, die verschiedene Arten von Verschlüsselung nutzen
- Zwei Protokolle haben sich durchgesetzt
  - TLS/SSL für HTTP Verbindungen
  - SSH für remote Logins

SSH



SSH

- Schutz vor dem Abhören von Verbindungen Abhören von Passwörtern
- Früher telnet (kein Schutz) heute SSH
- Verschlüsselte Verbindung, Passwort geschützt
- Asymmetrische Verschlüsselung zum Austausch von Sitzungsschlüsseln
- Sitzungsschlüssel sind symmetrisch

Problem von SSH

- Abgesehen von den üblichen kryptanalytischen Problemen
- Woher weiss ich, dass ich mit dem richtigen Rechner verbunden bin?
- Schlüssel selber vergleichen: schwierig bei langen Schlüsseln und nur diese sind sicher
- Lösung: Fingerprint des geschickten Schlüssels wird generiert
- Der Benutzer vergleicht den Fingerprint mit ihm bekannten Fingerprint

Demo SSH

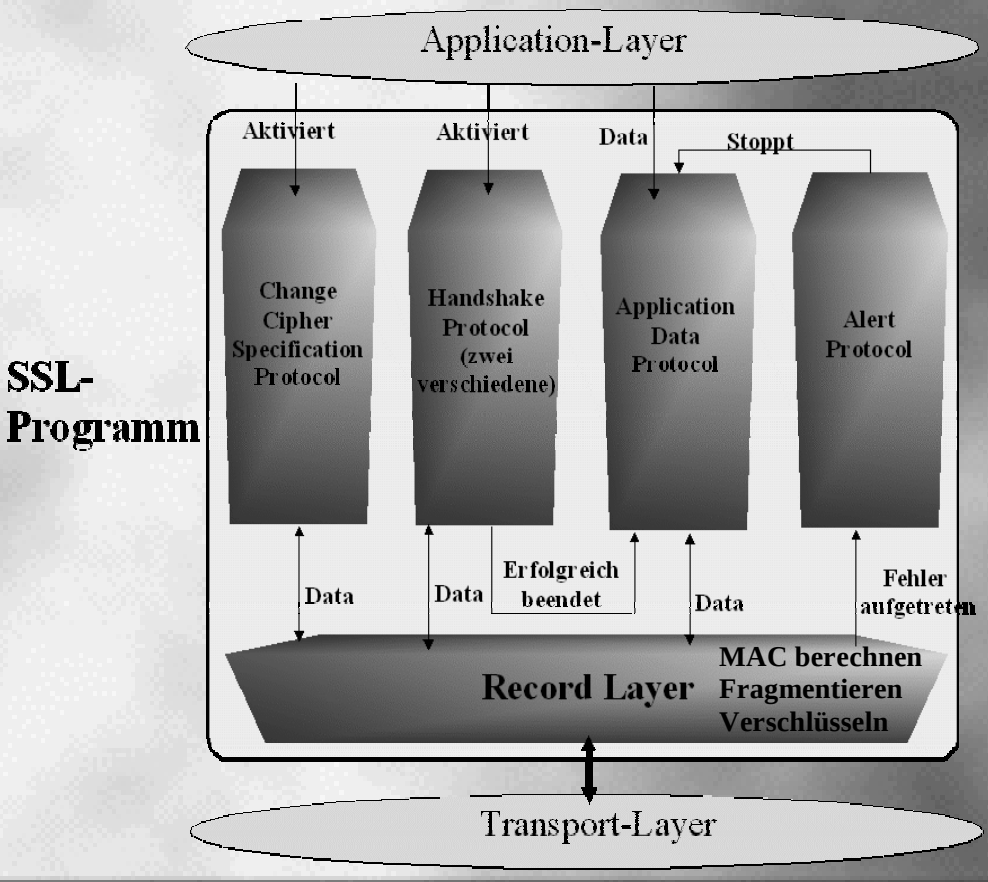
Wieso SSL?

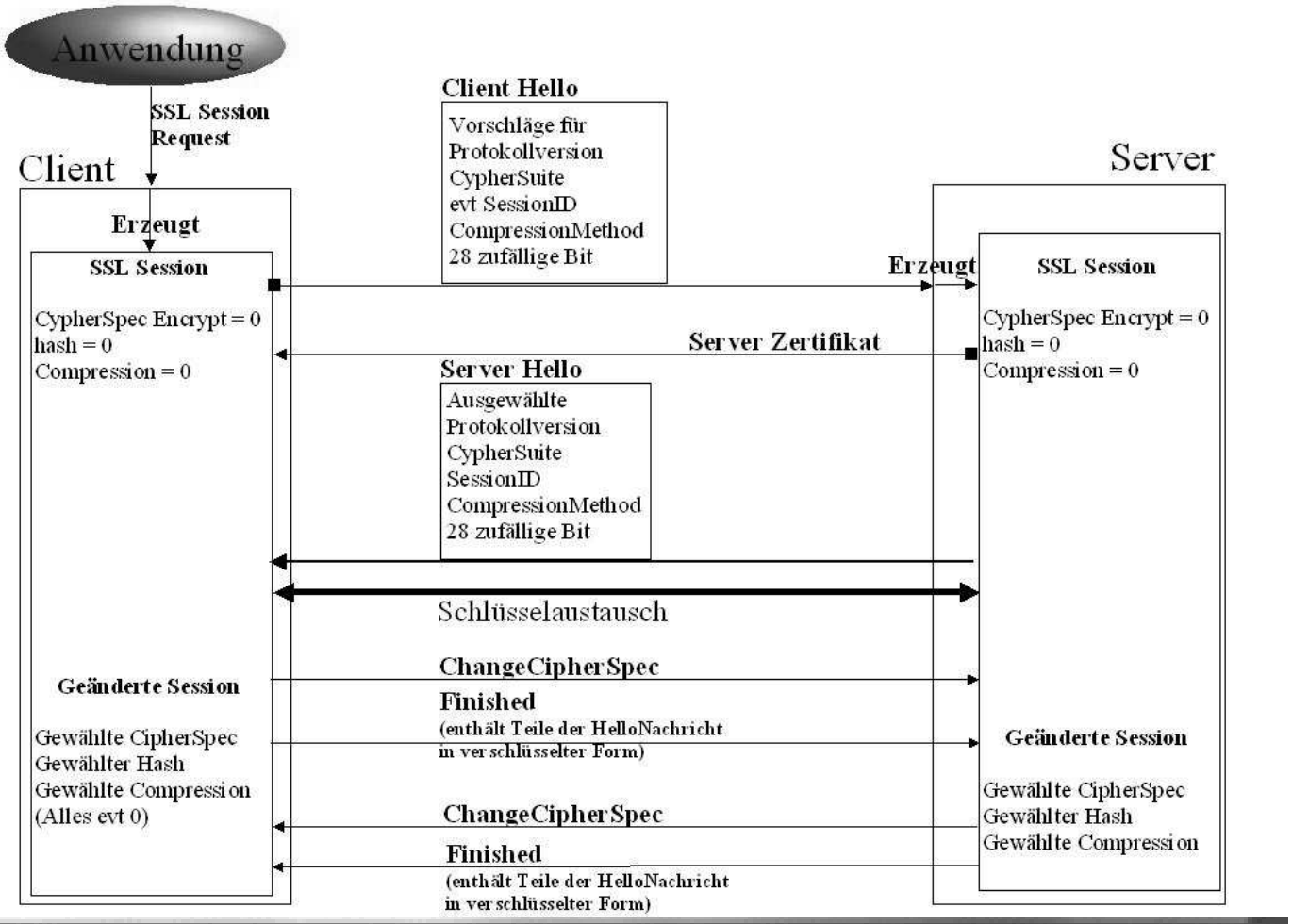
- Manchmal schickt man Daten an Webserver, die geheim bleiben sollen (Beispiel: Online-Banking per Browser)
- Lösung: Verbindung wird verschlüsselt
- Problem: Ich möchte Server kontaktieren, deren Schlüssel ich nicht kenne (auch nicht den Fingerprint). Es ist zu umständlich, sich alle Schlüssel aller Server zu beschaffen. Schlüssel ändern sich: und dann?
- Problem: auch unkundige Benutzer haben ein Recht auf Sicherheit!

Deshalb SSL

- SSL baut Verbindungen transparent für den Benutzer auf
- SSL prüft, ob der Verbindungspartner authentisch ist (gegen Man-In-The-Middle Angriff)
- Alles ganz einfach?!

Funktionsweise von SSL



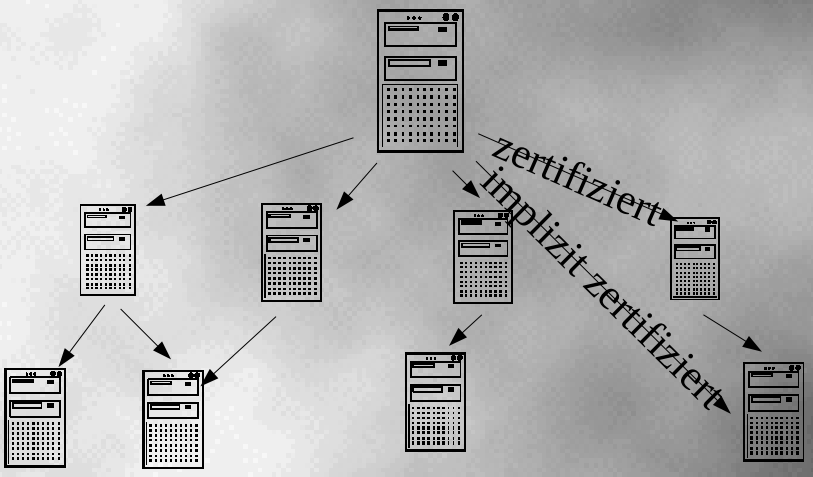


SSL

- Öffentliche Schlüssel, ähnlich wie bei der von PGP verwendeten PKI
- Mit Hilfe der öffentlichen Schlüssel werden dann Sitzungsschlüssel ausgehandelt
- Woher weiss ich, dass der Schlüssel authentisch ist?
- Erinnerung: jeder soll Surfen können, nicht nur Experten (der Schlüssel soll automatisch getestet werden)
- Lösung: Schlüssel werden von zentralen Instanzen zertifiziert (signiert), deren Schlüssel mit dem Browser installiert wird, so dass die Zertifikate testbar sind
- Schlüssel werden automatisch vom Browser getestet

Zertifikatsketten

- Das Zertifikat eines Servers wird von einer vertrauenswürdigen zentralen Instanz signiert.
- Der Browser testet das Zertifikat auf Gültigkeit, ohne dass der Benutzer es merkt



Probleme von SSL

- Vertraue ich der zentralen Instanz wirklich?
- Art des Schlüsselaustausches, Verschlüsselungsverfahren etc werden transparent für den Benutzer ausgehandelt. Was ist, wenn ich IDEA will?
- Die Zertifikatsprüfung bleibt der Willkür der Browserprogrammierer überlassen. Woher weiss ich, dass mir nicht nur ein gültiges, sondern auch ein passendes, von einer vertrauenswürdigen Stelle signiertes Zertifikat geschickt wurde?

Demo IE

Unnu?

- Ok, damit hätte ich die Verbindungen gesichert, die ich nach draussen aufbaue
- und der Rest? Was kommt da noch?
  - Sachen, dei mir geschickt werden: IE, Netscape
  - Betriebssystemsicherheit *aua*
  - Viren etc
  - Eingehende Verbindungen..

Firewalls

- Manche der erwähnten Scheisserchen (Trojaner, Viren etc) bauen Verbindungen nach draussen auf, ohne dass ich das initiiere, merke, oder geschweige denn, will...
- Es gibt Programme und Menschen, die suchen im Internet gezielt nach Rechnern mit Schwachstellen. Dazu sprechen sie (zB) einfach eine IP nach der anderen an: wenn sich ein Rechner mit bestimmten Angaben meldet (Protokolle geben ja Infos), dann ist eine Schwachstelle gefunden. Heureka!

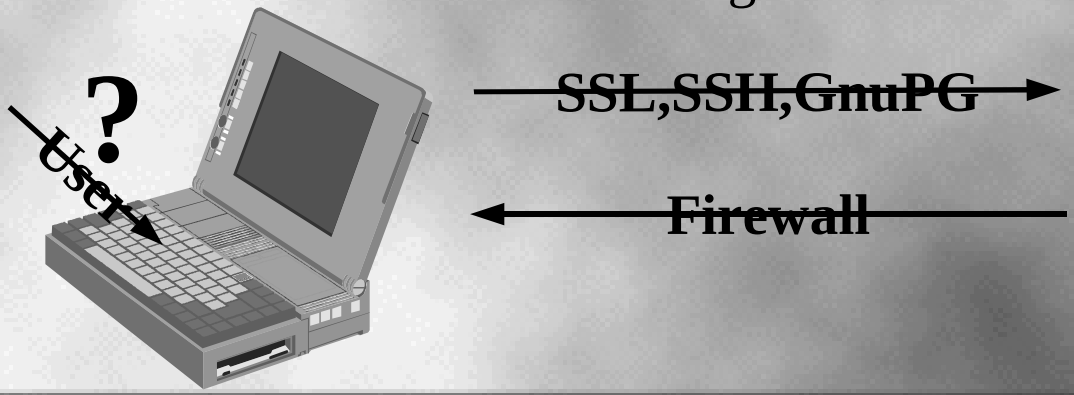
Firewalls

- Firewalls kontrollieren eingehende und ausgehende Verbindungen.
- Manche Firewalls können Viren filtern und Angriffe erkennen (Ressourcenaufwendig und teuer, für Gateways)
- Personal Firewalls arbeiten auf Programmebene: möchte man einem Programm erlauben, das Internet zu kontaktieren? Möchte man einen Kontakt akzeptieren?
- Firewalls, die auf Portebene arbeiten, sind für private Anwendung nicht gut geeignet. Wieso?
- Manche Firewalls starten erst nachdem sich der Benutzer angemeldet hat: bei dauerhaften LAN Verbindungen: aua!

Demo Zone Alarm

Gefahren im Internet

- Durch die gezeigten Massnahmen kann man sich gegen Abhören von Informationen und Passwörtern einigermassen schützen
- Eingehende und ausgehende Verbindungen werden kontrolliert. Oder nicht?
- Was für weitere Gefahren gibt es?



Gefahren durch den User/PC selbst

- Aufspielen von verseuchten Programmen
  - Trojaner
  - Viren
  - Dialer
- Schwachstellen im PC/Betriebssystem, in den Programmen
  - Windows
  - andere fehlerhafte Programme
- SPAM (Kein Virus, aber trotzdem Schaden)
- Schlechte Passwörter
- ...

Trojaner

- Trojaner sind Programme, die sich als harmlos und nützlich tarnen, dabei aber Informationen sammeln und sie an Server weiter geben (Spyware) oder anderen Quatsch machen
  - Manche Trojaner kontaktieren von sich aus einen Server
  - Manche Trojaner warten darauf, dass sie kontaktiert werden, dazu lauschen sie an Ports
- Trojaner können auch Hintertüren auf dem PC öffnen, durch die ein Hacker dann freien Zugriff auf die Ressourcen gewinnt

Trojaner - Fortsetzung

- Trojaner kann man sich, wie auch Viren, einfangen, indem man harmlos aussehende Programme installiert
- Spyware wird oft freiwillig installiert, zB weil nur mit ihnen zusammen nützliche Software aktiviert werden kann (zB Gator)
- Spyware sammelt zB Passwörter oder fängt generell Tastendrücke ab

SpyBot Demo

Spybot erkennt Trojaner, zusätzlich hilft eine Personal Firewall, die unerwünschte Kontakte verhindert

Viren

- Viren kann man sich durch das starten/installieren verseuchter Programme oder Dateien einfangen (exe, doc, ... alles was ausführbaren Code enthält!)
- Es gibt Unmengen verschiedene Viren, täglich neue
- Kostenlose Virenbaukästen erleichtern den „Neubau“ von Viren
- Viren starten sich automatisch selbst und hängen dann versteckt im Speicher herum, von wo sie andere Programme infizieren und sonstiges Unheil anrichten

Virenscanner

- Virenscanner suchen nach Signaturen der Viren
- Signaturen sind Bitfolgen im Code, durch den sich Viren auszeichnen oder andere Merkmale, die für bestimmte Viren immer gleich sind
- Manche Viren tricksen Scanner aus
- Manche Virenscanner bleiben deshalb im Speicher und achten permanent auf verdächtige Verhaltensweisen
- Gute Scanner suchen auch nach Heuristiken: bestimmte Signaturen sind typisch für Viren, aber nicht nur für Viren

Benutzung von Virenscannern

- Anleitung lesen!
- Virensignaturen auf allerneuesten Stand bringen: meistens von den Webseiten der Hersteller runtezuladen.
- Neuest erhältliche Version des Scanners benutzen: neue Viren kommen stündlich „auf den Markt“
- Bei Virenverdacht: am besten Betriebssystem von CD oder Diskette starten (Virenfreies!) und von dort aus scannen. Nur dann hat man Gewissheit, dass der Virus den Scanner nicht austricksen kann.

Virenscanner benutzen Teil2

- Evt im abgesicherten Modus starten (da werden keine Dienste automatisch gestartet) und alle Autostartmöglichkeiten absuchen
- Es nützt oft nichts, von Disketten zu booten, die auf dem jetzt befallenen Computer benutzt wurden oder einen Virenscanner zu installieren. Viele Viren halten sich lange bedeckt, ehe sie ausbrechen: die Diskette könnte verseucht sein. Der Virenscanner wird evt vom Virus erkannt und geblockt.

Vorsichtsmassnahmen

Durch einige Vorsichtsmassnahmen kann man sich präventiv gegen Virenbefall schützen

- Keine fremden Disketten benutzen
- Niemals auf Dateianhänge klicken, die per Email geschickt werden, ohne sie vorher gründlich auf Viren zu scannen
- Kein Email Programm verwenden, dass Dateianhänge in irgendeiner Form ausführt
- Ggfs alle Ausführ-Optionen im Emailprogramm deaktivieren
- Sicheres Mailprogramm verwenden (ZB pine)

Weitere Vorsichtsmassnahmen

- Keine Programme aus dem Internet saugen, mindestens auf Viren scannen vor der Installation
- Bei Virenverdacht: Windows sofort! per Stecker raus beenden! und im abgesicherten Modus starten. Mit autoruns alle Programme kontrollieren und ggs deaktivieren, die sich automatisch aufstarten.
- Alternativ mit einem garantiert Virenfreien Betriebssystem booten und Viren scannen!

Demo autoruns

Demo Virenscanner

- Bei Benutzung und Installation Vorsichtsmassnahmen beachten: manche Viren erkennen Scanner und tricksen sie aus, wenn man sie im Speicher hat
- Immer aktuelle Virendatenbanken besorgen! Kriegt man beim Hersteller derVirenscanner.
- Anleitung lesen ist hier eine extrem gute Idee!
- Bei Virenverdacht: möglichst Viren ausschalten oder alternativ mit garantiertg sauberem Betriebssystem booten!

Dialer

- Dialer sind Programme, die sich unbemerkt über das Modem ins Internet einwählen
- Statt der gewünschten Verbindung über einen Provider können zB 0190 Nummern gewählt werden
- Dialer Control findet Dialer anhand von Signaturen: Programmteile, die typisch sind für Dialer
- Dialer Control bleibt auch im Speicher und achtet darauf, ob ein Programm auf das Modem zugreift. Ähnlich wie bei ZoneAlarm kann man den Zugriff programmbasiert erlauben oder sperren

Demo Dialer Control

Remote Administration Programs

- Remote Administration Programme sind solche, die die Administration, also Kontrolle eines PCs über das Netz hinweg erlauben
- nicht unnütz, erlauben sie doch, Hilfe durch erfahrene Benutzer über's Netz
- ABER: sie erlauben auch Angreifern, sich einzuloggen und fröhlich Quatsch zu machen
Jeder Angreifer wünscht sich so ein Tool auf deinem Rechner!

Remote Administration Programs

- Besonders hervorgetan haben sich: Bac Orifice und Netbus
- Finden und Deaktivieren!
- Verraten sich dadurch, dass sie Versuchen, als Server zu agieren. Sie sollten also von der Firewall gemeldet werden.

Schwachstellen im Betriebssystem

- Ausnutzen von Schwachstellen im Betriebssystem
  - Durch Portscans stellt ein Rechner fest, ob mein PC verwertbare Angriffspunkte bietet
  - Danach kann er diese Punkte ausnutzen, mir zB einen Virus oder einen Trojaner einzuschleusen
  - Viren/Würmer finden und Nutzen Schwachstellen automatisch
- Schutz
  - Aktuelle Patches einspielen
  - Personal Firewall
  - Windows ordentlich konfigurieren!

Windows

- Am sichersten: deinstallieren und Linux benutzen
- Auf Windows XP, 2000 oder NT umsteigen
- Patch as Patch can!
- Gastaccounts/Serviceaccounts zu!
- Passwörter vergeben
  - Gute Passwörter
- Automatische Kontakte zum Web deaktivieren:
  - Service-Ports
  - Automatische Updates

Windows 95/98

- Kaum zu sichern: Win95/98 benutzen ein veraltetes Filesystem (FAT/FAT32), dass keine Zugriffsrechte für Dateien oder Verzeichnisse kennt

Patchen

- Patches in der richtigen Reihenfolge einspielen!
- Neuester Patch (Service Pack) genügt meistens
- Service Packs sind akkumulierte Patches und beheben eine ganze Reihe von Problemen parallel
- Wenn man warten kann: den SP nicht am ersten Tag aufspielen, sondern Erfahrungen anderer Nutzer abwarten
- Ggfs bei www.cert.org auf Hinweise achten

Ordneroptionen

- Einfache Freigabe abschlaten (Freigabe ohne Passwort)
- Alle Dateien und Pfade anzeigen
- Endungen für alle Dateien anzeigen!
- Arbeitsplatz|Extras|Ordneroptionen|Ansicht

Fehlerberichte

- Keine Fehlerberichte an Microsoft senden
- Irgendwie Geschmacksfrage: ich aber möchte NICHT, dass Microsoft alles über meine Installation weiss
- Arbeitsplatz|Rechte Maustaste|Eigenschaften|Erweitert

Zugriffsrechte auf neueren Windows Versionen setzen

Voreingerichtete Accounts

- Manche Windows Versionen haben nach Neuinstallation, also sozusagen ab Werk, Accounts mit Defaltpasswörtern aktiviert
- Das können Gastaccounts oder Administrator Accounts sein.
- Die Passwörter der voreingestellten Konten sind immer gleich und allgemein bekannt
- Diese Accounts deaktivieren!
- Start|Systemsteuerung|Benutzerkonten

Accounts bearbeiten

Automatische Aktualisierungen

Manche Programme, u.a auch Windows, versuchen automatisch und oft ohne Nachfrage, sich zu aktualisieren.
Das sollte man tunlichst unterbinden, da kaum nachvollziehbar ist, welche Daten dabei mitgeschickt werden. Ausserdem verlangsamt es unter Umständen (Bsp realplayer) die Internetverbindung unerträglich

- Firewall
- Programme aus dem Autostart werfen
- Windows: Arbeitsplatz|RM|Eigenschaften|Automatische Updates

Remote „Unterstützung“

- Remote Unterstützung deaktivieren
- Arbeitsplatz|RM|Remote
- !!!

Offene Ports

- Windows selbst aktiviert beim Starten einige Dienste und lauscht als Server auf Ports
- Firewall: es scheint nicht möglich, diese Dienste zu deaktivieren (Anfrage nach Freigaben und Angaben über den Benutzer werden über zwei Service-Ports geliefert)

Freigaben

- Manche Windowsversionen geben Ressourcen (Festplatten oder Drucker) defaultmässig frei
- Überprüfen mit
net share
- Solche Freigaben werden bei manchen Windows Systemen automatisch bei Systeminstallation eingerichtet
- Freigaben nicht unterstützen
- Siehe folg Folie

Windowsspezifische Kommunikationsprotokolle

- Windows hat spezielle Kommunikationsprotokolle für Microsoft-Netzwerke. Diese sind nach Installation bzw Inbetriebnahme einer Netzwerkkarte oft Defaultmässig aktiviert.
- Sie bieten einem Angreifer ungeahnte Zugriffsmöglichkeiten
- Sofort Dekativieren, falls man sich nicht in einem Microsoft-Netzwerk befindet (selten)

Demo Freigaben und Netzwerkprotokolle deaktivieren

Schwachstellen in Programmen

- Viele, eigentlich die meisten, Programme haben Schwachstellen, die von Hackern ausgenutzt werden können, um an einen ansonsten geschützten Rechner heranzukommen.
- Als Benutzer kann man sich dagegen nicht schützen, wenn man diese Programme verwenden möchte.
- Einziges Mittel: immer alle Patches einspielen
  - Manche Patches reissen Sicherheitslücken auf
  - Schwachstellen werden nicht schnell genug erkannt/ gepatcht.

Ausnutzen von Schwachstellen in Programmen

- Portscan um bekannte Schwachstellen zu finden oder direktes Ansprechen eines Rechners auf die Schwachstelle: reagiert er, ist er angreifbar
  - Angriff mittels Bufferoverflow
  - Ausnutzen anderer Programmfehler
- Überrumpeln von Benutzern (Ausführen von Code, Social Engeneering)
- Danach:
  - Zugriff auf Ressourcen
  - Zerstörung/Störung oder Nutzung des PC als Zwischenstation für weitere Angriffe

Ausnutzen von Schwachstellen in Programmen

- Ausnutzen von Fehlern in Email-Clients/Browsern/Chat Programmen... allen Programmen, die das Interner kontaktieren.

Dagegen hilft auch keine Firewall, da der Kontakt ja gewünscht ist...

- Danch: Ausnutzen von Sicherheitslücken: lesen von Dateien, Einschmuggeln gefährlicher Programme, die weitere Hintertüren öffnen...

Browser sichern

- Auch hier gilt vor Allen Dingen: Patches einspielen, neueste Versionen verwenden
- Cookies: keine Gefahr für den Rechner aber evt Gefahr für die Privatsphäre
- Javascript und Java: Sandboxprinzip
  - trotzdem Probleme durch Fehler in der Programmierung
- ActiveX, ActiveScripting etc: mehr Komfort, aber mangelhafte Sandbox

Browser sichern - Fortsezung

- Browser bringen Schutzmassnahmen mit, die nicht durchschaubar sind (besonders IE) und oft fehlschlagen
- Alle aktiven Elemente deaktivieren ist am sichersten
  - Viele Webseiten nicht mehr bedienbar
  - Aktuelle Patches einspielen, ActiveX etc deaktivieren, nur das Nötigste aktiviert lassen
- Demo: IE Einstellungen

Demo IE Einstellungen Sicherheit

Social Engeneering Benutzer überrumpeln

- Eine beliebte und recht erfolgreiche Angriffsart ist es, den Benutzer glauben zu machen, er müsse
  - Zugriff auf den Computer gewähren,
  - ein Programm ausführen,
  - sein Passwort verraten
- Verraten Sie NIEMALS ihr Passwort: auch nicht einem Ihnen bekannten Superuser: der braucht es nicht!

Social Engeneering Teil2

- Geben Sie NIEMAND den sie nicht kennen, Zugriff auf den Computer, egal ob direkten pysikalischen Zugriff oder per login oder sonstwie, auch wenn er/sie behauptet, Techniker zu sein
- Führen Sie NIEMALS Programme aus, die aus unbekannter Quelle stammen
- Dazu gehören auch Programme und Dateianhänge, die per Email geschickt werden.
  Mittlerweile sollten Sie wissen, wie leicht man Emails und Absender fälschen kann!!!

SPAM

- SPAM ist unerwünschte Email
- Lästig wird sie durch die pure Masse
- Der Name kommt von einem Monty Python Sketch
- SPAM wird erkannt
  - Durch Filtern der Emails und erkennen von typischen Mustern: *FREE* :-)
  - Durch Testen der Absenderadresse: echt oder gefälscht?

SPAM erkennen

- Unbekannte Absender
- Inhalt
  - SPAM öffnen ist problematisch, weil einige HTML Seiten enthalten, die Grafiken laden. Schon daran erkennt der Versender des SPAM, dass sein Kram gut angekommen ist.
    ->HTML-Email abschalten!

Zustandekommen von SPAM

- Hoaxes (Viruswarnung: bitte an alle Adressen weitermailen!!)
- Kettenbriefe
- Webcrawler durchsuchen das Internet (Webseiten, News, etc) nach Emailadressen
- Email Adressen werden verkauft an Leute, die Werbung verschicken wollen
- Wenn nur jede 5000te SPAM-Mail beantwortet wird, lohnt sich die Sache: Spammen ist billig!!!
- Gesetze gegen SPAM sind nur bedingt hilfreich Die Spammer sind oft schwer auffindbar

SPAM verhindern

- SPAM kann man ausfiltern
  - Nachteil: auch erwünschte Emails können herausgefiltert werden
- Benutzung von 2 Adressen: eine öffentlich gemachte, eine für Freunde und Bekannte
  - auf der öffentlichen kriege ich genauso viel SPAM wie sonst auch, ausserdem, wenn die private mal auffliegt, habe ich doppelt so viel SPAM

SPAM verhindern

- Ich verrate meine Email niemand
  - Wozu dann 'ne Email?
- Email verschlüsseln: benutzerspezifische Mailadressen
(da muss der ServerAdmin mitmachen und das entsprechende Programm zur Verfügung stellen und unterstützen)

SPAM verhindern

- Email Adresse auf der Webseite „verschlüsselt“ angeben. ZB auf ein Blatt Papier schreiben, einscannen und als Grafik auf die Webseite.
  - Schützt 100% gegen Webcrawler, Email lässt sich nicht mehr klicken, aber...
- Wenn die Adresse mal bekannt ist: Filter, entweder direkt auf dem Server oder auf dem Rechner (Internetverbindung dann trotzdem zugemüllt)
- NIEMALS auf SPAM antworten

Gute Passwörter

- Viele Windows Rechner sind per Login legal von Aussen zugreifbar
- Passwortangriff: ein Hacker versucht sich Zugang zu einem Rechner zu verschaffen, indem er ein Passwort rät
- Gutes Passwort:
  - Kein Wort aus irgendeinem erreichbaren Lexikon
  - enthält nicht nur Buchstaben
  - gut merkbar und sicher: Ganze Sätze, aber nur die Anfangsbuchstaben (Endbuchstaben) ins Passwort GSandAEiP

Genereller Ablauf eines Angriffs

- Schwachstelle finden (User, Programme, Betriebssystem)
- Zugriff auf Ressource gewinnen (Datei lesen und Passwörter raten/knacken, Backdor einschmuggeln)
- Vollzugriff auf den Rechner: Quatsch machen, wietere Rechner hacken, erstmal stille sein, später DOS Angriff auf anderes System...

DoS - Denial of Service

- Denial of Service Angriffe sind solche, die darauf abzielen, eine Ressource, d.h. einen Rechner oder einen Teil des Netzes lahm zu legen.
Sehr oft, indem ein Rechner (zB ein Server) so mit (falschen) Anfragen zugeknallt wird, dass er keine Ressourcen mehr hat, echte Arbeit zu erledigen
- Distributed Denial of Service: eine Ressource (Rechner) wird von vielen verschiedenen Computern zugleich angegriffen

Schutz - Vorgehen

- Windowskonfiguration auf Sicherheitslücken prüfen (Benutzer, Backdoors, Protokolle...)
- msconfig
- Virenscan (regelmässig, sowie on-access)
- AntiDialer, DialerControl
- regelmässig AntiSpyware anwenden
- Sicheres Email-Programm
- Browser sichern
- SSL, SSH, GnuPG verwenden

Literatur

- http://www.cert.org
- Applied Kryptography, Bruce Schneier
- Computer Networkin, Kurose and Ross
- Hackers Guide, anonymous
- c't
- http://www.bsi.bund.de
- http://www.heise.de

Danke für eure Aufmerksamkeit!

- Fragen?
- CDs verteilen
- Kopien verteilen
- Abschied :-(