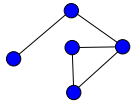
Gliederung

Peer-to-peer Systeme und Datenbanken(SS07)

- Kapitel 1: Einführung
- Kapitel 2: Beispiele
- Kapitel 3: Routing
- Kapitel 4: Schemabasierte p2p-Netzwerke
- Kapitel 5: Integrationsprobleme
  Teil 5-1: Einführung, Gleichheit
  Teil 5-2: Ähnlichkeit - 1
  Teil 5-3: Ähnlichkeit - 2
  Teil 5-4: Mappingbasierte Datenintegration
- Kapitel 6: Anonymität, Authentifikation
- Kapitel 7: Reputation

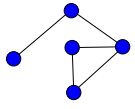
Version vom 22. Juni 2007



Kapitel 6

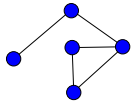
Gliederung:

- Motivation
- Authentifizierung
  mit und ohne Notar
  Gruppenzugehörigkeit
  Authentifikation einer Nachricht, Bestätigung des Empfangs.
- Anonynität
  Anonymität der Kommunikation
  Anonynität der Subjekte
  Projekte zur Wahrung von Anonymität



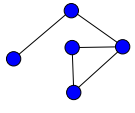
Warum?

- Funktionalitätsklassen F7 - F9: Funktionen sicherer Systeme in Netz
  - Wechselseitige Identifikation und Authentifizierung
  - Unbestreitbarkeit der Sendung bzw. des Empfangs
  - Sichere Übertragung über unsicheren Kanal
  - Verbergen von Absender, Empfänger, Tatsache der Übertragung
- Voraussetzung für geschäftliche Akzeptanz/Nutzung
- Defizite bei existierenden Systemen, am ehesten noch Ansätze von Anonymität bei einigen Tauschbörsen.



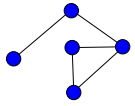
Identifikation

- Wechselseitige Authentifizierung in reinem P2P-Systemen anfällig gegen *man-in-the-moddle*-Angriff,
  dieser ist grundsätzlich in allen Protokollen möglich, die nicht mit geheimen Informationen arbeiten.
  Ausweg:
  - Systeme mit Notar (d.h. hybride Systeme): es muß eine initiale sichere Kommunikation mit Notar geben
  - Systeme mit übertragbarem Vertrauen
- Symmetrische Schlüsselsystem scheiden aus (Komplexität)



Identifikation mit Notar T

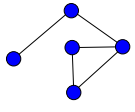
- Neuer Peer P weißt auf sicherem Weg dem Notar seine Identität nach und erhält von ihm: seinen privaten Schlüssel, seinen öffentlichen Schlüssel mit seiner Identität und der Signatur des Notars, den öffentlichen Schlüssel des Notars. Der Notar behält den öff. Schlüssel von P (mit Zuordnung zu P).
- P will mit Q kommunizieren:
  - Anforderung des öff. Schlüssel von Q bei T (signiert, verschlüsselt) - wird verschlüsselt übermittelt. P kennt jetzt Q.
  - P sendet an Q verschlüsselt: Seine Identität und seinen ö. Schlüssel.
  - Q öffnet mit seinem p. Schlüssel, prüft Id: Q kennt jetzt P
- Hybrides System, Notar evt. Engpass, Sicherheit des Systems = Sicherheit und Vertrauenswürdigkeit des Notar.



Identifikation mit Notar T -Echolink

k4.tex

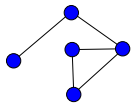
- Kontakt mit Notar:
  Fax oder e-mail: Name, Vorname, Rufzeichen der Station, Kopie der Lizenzurkunde der nationalen Fernmeldebehörde, aktuelle e-mail-Adresse
  Betreiber wirkt als Notar, vergleicht Angaben mit DB bzw. Listen der nat. Behï£¡den und weiteren Quellen $\mapsto$ Peer erhält e-mail mit Paßwort
  Anderungen erfordern evt. neuen Nachweis
- Diese Authentifikation ist täuschbar, dies ist aufwendig, da viele Details gegeneinander geprüft werden, die alle öffentlich sind. Wird von vielen nat. Fernmeldebehörden akzeptiert



Identifikation ohne Notar

k4.tex

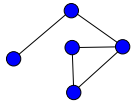
- Voraussetzung: alle Peers sind vertrauenswürdig, alle Kanäle im P2P-Netz sind sicher.
- Anmeldung: neuer Peer Q authentifiziert sich gegenüber einem beliebigen Peer P. P behandelt diese Information wie jeden anderen Inhalt und gibt sie auf Anfrage heraus bzw. baut eine DB bekannter ID auf.
- Login von Q: bei belieb. Peer R, dieser befragt seine DB, falls negativ, Anfrage an andere Peers, kommt eine positive Antwort, wird Q akzeptiert.
- nur einseitig, Q hat keine Kontrolle, mit wem er kommuniziert; P2P-Netz erscheint wie ein Computer.
- Variante (ähnlich pub.Key-Systemen): nicht jeder gilt als völlig vertrauenswürdig, jeder Peer gibt eine positive Antwort nur weiter, wenn er sie von einem Peer bekommt, dem er ausdrücklich vertraut.



Anonyme Gruppenzugehörigkeit

Nachweis, daß ein Peer P zur Gruppe der Peers gehört, ohne daß er seine Identität preisgibt.

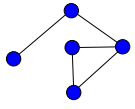
- scheinbarer Widerspruch der Anforderungen
- Literatur:(Auswahl)
  K.Koyama, „Demonstrating Membership of a Group Using the Shizuya-Koyama-Ithoh (SKI) Protokol“, Proceedings of the 1989 Symposium on Cryptography and Information Security (SCIS 89), Gotenba, Japan, 1989
  C. Shu, T. Matsumoto, and H. Imai, „A Multi-Purpose Proof System, Transactionsof the Institute of Electronics, Information, and Communication Engineers, v. E75-A, n. 6, Jun 1992, pp. 735-743
- eine triviale Idee: Es gibt im P2P-Netz ein Schlüsselpaar (p,ö), das jedes Gruppenmitglied (und nur ein solches) kennt. Das wird zu einer wechselseitigen Authentifizierung nach einem Protokoll für unsymmetr. Kryptosysteme benutzt.



Gruppenzugehörigkeit mit Identifikation

Kommutative Einweg-Akkummulatoren
Literatur: J.C. Benaloh and M. de Mare, „One-Way Accumulators: A Decentralized Alternative to Digital Signatures “, Advances in Cryptology - EUROCRYPT 93 Proceedings, Springer-Verlag, 1994, pp.274-285aten werden nach (m,n)-Verfahren geteilt und auf Peers verteilt

- Die Liste der Mitglieder ergibt bei beliebiger Permutation den gleichen Hashwert
- Jedes Mitglied berechnet den Hashwert, läßt sich dabei außer Acht.
- Zwei Mitglieder, die sich identifizieren, nennen jeweils Ihren Hashwert und Namen und rechnen die erhaltenen Daten nach.
- Andere Mitglieder nicht betroffen, neue Mitglieder hinzufügbar, Streichung: neue Liste

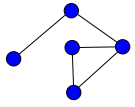


Schlüsselaustausch

in einem unsicheren Kanal

- - P, Q vereinbaren große Zahlen n, x (öffentlich)
  - P merkt sich p, Q q (geheim)
- - P: $x_1 = x^p \mod (n)$ an Q,
  - Q: $x_2 = x^q \mod (n)$ an P
- - P: $z_1 = x_2^p \mod (n)$
  - Q: $z_2 = x_1^q \mod (n)$

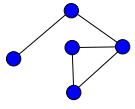
  P und Q kennen $z_1 = z_2$.
- Das Berechnen des Logarithmus einer Zahl in einem Restklassenring gehört zu den schweren Problemen.
- keine Authentifikation



Authentifikation einer Nachricht

Nachweis, daß die Nachricht wirklich vom Absender stammt

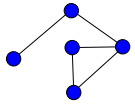
- Symmetr. Verfahren ohne Notar: Empfänger kennt dem Absender, kann dies aber keinem Dritten beweisen.
- Symmetr. Verfahren mit Notar, der eine DB über alle Zustellungen führt. Da er das Vertrauen aller hat, wird seine Entscheidung akzeptiert.
- Unsymmetr. Verfahren: Signatur mit privatem Schlüssel, jeder kann mit dem öffentlichen Schlüssel des Absenders die Signatur prüfen.
- zusätzlich Zeitstempel, Gültigkeitsintervalle der Signatur sichern gegen wiederholte Verwendung, Verschlüßelung gegen Verlust an Vertraulichkeit.



Bestätigung des Empfangs

Setzt Kooperation des Empfängers voraus

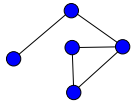
- Empfänger Q prüft die Korrektheit der empfangenen signierten Nachricht, bestätigt den Empfang und signiert das Paar (Nachricht mit Absender-Signatur, Empfangsbestätigung) und sendet das alles an Absender.
- Alle Übertragungen werden mit dem ï£¡fentlichen Schlüssel des jeweiligen Empfängers chiffriert.
- Angriffe möglich, deshalb verschiedene Schlüsselpaare für Chiffrierung und Signierung benutzen.



Bestätigung des Empfangs (2)

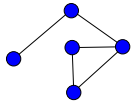
Nicht vertrauenswürdige Partner
Durch Kombination mit Geheimnisteilung mit bitweisem Austausch der Information wird Betrug verhindert.
Anschaulicher Vergleich: Zwei Parteien unterschreiben ein Dokument ohne Notar abwechselnd und buchstabenweise

- Oblivious Transfer (nichteindeutige ï£¡ertragung)
  Bruce Schneier: Angwandte Kryptographie, Add.Wesley, S.145
  Die Nachricht wird mit einem symmetrischen Verfahren verschlüsselt, der Sitzungsschlüssel wird bitweise im Bestätigungprotokoll übergeben und im Gegenzug wird die Bestätigung der Nachricht bitweise bestätigt. Eine Verquickung mit n Schlüsseln sorgt dafür, daß keine der beiden Parteien betrügen kann.
- Inhalt bzw. der Wert der Nachricht wird nicht gesichert.
- Problem in P2P-Netzen: viele Nachrichten mit wenig Information



Geheimnisteilung

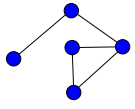
- Idee: Eine Nachricht m wird in N Teile zerlegt, nur alle Teile zusammen gestatten die Nachricht zu rekonsruieren.
- M-N-Schwellwertverfahren: M von N Teilen reichen, die Nachricht zu rekonstruieren
- 2 Teile: s Zufallsstring, $t = m$ xor s
- M-N-Verfahren: m wird in M Teile zerlegt: x_i, i=1,...M.
  Diese dienen als Unbekannte in einem überbestimmten Gleichungsystem mit M Unbekannten und N Gleichungen, von denen je M linear unabhängig sind.
- Redundanz, wenn alle Teile auf verschiedene Peers verteilt werden.
  Beitrag zur Verfügbarkeit, zur Anonymisierung.



Annymität

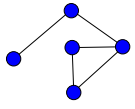
- Reaktion auf Zensurversuche im Internet.
- Das Internet ist nicht anonym !

  Misbrauch eines Rechners in einem Firmennetz zum illegalen Filesharing -
  8 Stunden später Beschwerde des Geschädigten aus den USA eingetroffen.
- Gefahr des Mißbrauchs der Anonymität.
- Es gibt keinen absoluten Schutz, die Verfahren erschweren lediglich die Erkennung
- Methoden: Verschlüsselung, daß Inhalte verborgen, ... $\mapsto$ Vorlesung „Kryptographische Protokolle“
- Literatur: Dingledine,r. u.a.: *The Free Haven Projekt: Distributed Anonymos Storage Service.*



Anonymität

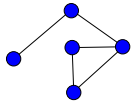
- Anonymität der Übertragung: Ein Dritter kann nicht erkennen, wer welche Nachricht sendet (A. des Senders), wer sie empfängt (A. des Empfängers) und ggf. auch nicht, daß überhaupt eine Nachricht gesendet wurde (A. der Übertragung).
- Anonymität in einem P2P-System zum Dokumententausch:
  - *A. des Autors*: (Für einen Außenstehenden) ist es unmöglich, einen Autor mit einem bestimmten Dokument in Verbindung zu bringen.
  - *A. des Verlegers*: wie eben, Verleger statt Autor.
  - *A. des Lesers*: wie eben, Leser statt Autor, Schutz der Privatsphäre des Lesers
  - *A. des Servers*: wie eben, Server statt Autor. Kennt ein Beobachter einen Dokument-Identifikator, kann er nicht ermitteln, wo sich das Dokument befindet.
  - *A. des Dokumente*, *A. der Anfragen*



Anonymität (3)

- *Anonymität des Dokuments*
  Server hat keine Information, welche Dokumente er hat. Schutz des Serverbetreibers, falls ein Beobachter sich Zugang zum Server verschafft.
  Passiv-Server Dokument A.
  Server sieht nur Daten, hat keine Information über Inhalte (z.B bei Geheimnisteilung, bei Verschlüsselung, wo Server auf den Schlüssel nicht zugreifen kann. *Aktiv-Server Dokumente A.*
  Server kann Daten mit anderen Servern abstimmen und Anfragen an andere Server stellen. Es braucht täuschungssicheren Mechanismus, der Serveranfragen von Clientanfragen unterscheidet.
- *Anonymität der Anfrage* Server kann nicht feststellen, welches Dokument er liefert, wenn er eine Anfrage bearbeitet.

Punkte nicht unabhängig voneinander.

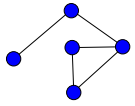


Anonymität der Übertragung

Ideen:

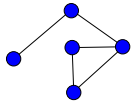
- Rundsprüche: unspezifizierte Adresse, jeder kann Nachricht empfangen, wenn mit öS E des Empfängers verschlüsselt, erkennt nur er den Inhalt.
  Schutz des Empfänger, nicht des Senders, hohe Netzlast.

- Nachrichten über Proxies, kann auch Sender anonymisieren (David Chaum: Mixes, 1981)
  A sendet Nachricht m an B über die Knoten $K_1, K_2, ..., K_n$ mit den Schlüsselpaaren $(E_i, D_i, i = 1...n)$: A:
  $(E_1, (K_2, E_2(...(K_n, E_n(m))...))) \mapsto K_1$, K_1 dechiffriert mit D_1, sendet weiter an K_2, usf., K_n ist B
  Alle Knoten außer K_n kennen m nicht; einzelne Nachricht nicht verfolgbar, wenn es (i) viele Nachrichten, (ii) viele Knoten und (iii) immer wechselnde Ketten gibt.
  Wertung für P2P: geringere Netzlast, Knoten einer Kette müssen verfügbar sein.



Pseudonyme

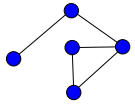
- Ein Pseudonym ist ein Attribut im Metadatensatz, dessen Werte jeweils genau einem Wert eines durch Anonymität verdeckten Attributs zugeordnet werden kann.
  Beziehung: Pseudonym : Attribut = N:1
  Das anonyme Attribut bleibt geschützt, aber alle Dokumente eines Pseudonyms als zusammengehörig erkennbar.
- Beispiel: Namenspseudonym : Autorname.
- Realisierung: z.B. Signatur eines Dokuments mit privatem Schlüssel.
- Bewertung der Reputation eines Herausgebers bei Wahrung der Anonymität.
- Grad der Anonymität. Kein Schutz perfekt. Kann Beobachter indirekt auf die wahre Identität schließen? (IP-Nummer, Bandbreite, ...)



Das Free Haven Projekt FHP

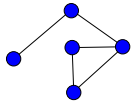
Literatur: Dingledine, R. u.a.: *The Free Haven Project: Distributed Anonymos Storage Service.* A.a.O.

- Designziele: Schutz der Anonymität und Widerstand gegen Angriffe
- Vergleichsmodell: A kommuniziert mit B über Notar T.
  Notar - nicht nur Vertrauensperson, Dienstanbieter, der Eigenschaften garantiert (Nachricht wird übertragen, Antwort ist möglich, Information gelangt nicht an Dritte) und keine undokumentierten Funktionen hat.
- 3 Rollen: Verleger, Server, Leser.
- Adressen im Applicationsnetz: remailer reply blocks (verschlüsselte Routing Informationen)
- Wegen broadcast-Suche mangelnde Effizienz.



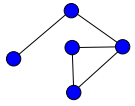
FHP - Dokument einbringen Suchen

- Autor sucht geeigneten Server / betreibt selbst Server.
- Datum $f \mapsto f_1, f_2, ..., f_n$, (k, n)-Geheimnisteilung (k von n)
  Dokumentspezif. Schlüsselpaar $(\ddot{o}S_{doc}, pS_{doc})$ erzeugt, f_i mit pS_{doc} signiert.
  (wenn k groß: eher Verlust von f möglich, k klein: große Teile.
  Einbringen jedes Teils (f_i +
  Metainformation($Verfallszeitpunkt, (k, n), \ddot{o}S_{doc}, Signatur$))
  Dokumentidentifikation Hashwert $H(\ddot{o}S_{doc})$
- Suche:
  Leser generiert $(\ddot{o}S_{client}, pS_{client})$ und one-time remailer reply block.
  Anfrage: $(H(\ddot{o}S_{doc}), \ddot{o}S_{client}, reply - block)$ an alle dem Leser bekannten Serverknoten.



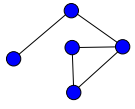
FHP: Anfrage bearbeiten, Gültigkeitszeit, Rückruf

- Jeder Server prüft, ob Daten zu $H(\ddot{o}S_{doc})$ bei ihm vorhanden,
  ja: Verschlüsseln mit $\ddot{o}S_{client}$ und Antwort via reply-block. Anfrager hat Daten, wenn er k (von n) verschiedene Teile hat.
- Gültigkeitszeit: Server kann Datum nach Ablauf der Gültigkeit löschen ohne Nachteile (für sich), dazu muß der Autor nicht aktiv werden (Schutz der Anonymität des Autors).
- Rückruf: Autor erzeugt geheime Zusatzzahl x und verteilt $H(x)$ mit den f_i. Sendet später Autor x an Server, ist das das Signal, die zu $H(x)$ gehörigen Daten zu löschen.
  Angriffe gegen Rückruf:
  - Keine Garantie, daß R. vollständig gelingt (Knoten nicht erreichbar, Knoten verweigert Dienst oder Weiterleitung von x).
  - $H(x)$ kann Urheber bloßstellen, wenn bei ihm x gefunden wird.
  - Angreifer ermittelt, wer x besitzt und erpreßt Rückruf.
  $\mapsto$ *Rückruf in FHP nicht realisiert*.



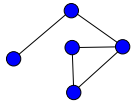
FHP: Trading - Dokumentenaustauch

- Gründe:
  Verbirgt Publikationstätigkeit u. Anfragen unter Tauschaktivität.
  Erlaubt geordnetes Verlassen des Netzes, indem Server nur kurzlebige Dokumente akzeptiert.
  Unterstützt lange Lebensdauer, es wäre sonst evt. schwierig, Server zu finden, die solche Dokumente akzeptieren.
  Server kann ihm mißliebige Dokumente wegtauschen (nicht vernichten - Reputation !!!). Wechselnde Inhalte: es gibt kein festes Ziel für Angriffe gegen ein Dokument.
- Tauschprotokoll: mehrstufige Verhandlungen, Einbeziehung eines *buddy* auf jeder Seite (Pauschpartner, Reputation?, Was wird getauscht: $Größe \times Lebensdauer$).
  Verletzt eine Seite das Protokoll: Nachricht des anderen an Reputationssystem, *buddy* Zeuge, verhindert auch Mißbrauch.
  Nach Tausch halten beide Seiten die alten Daten zeitweilig weiter für den Fall, daß der Partner doch betrügt.



FHP: Angriffe gegen Anonymität

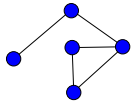
- beim Leser:
  (1) Angreifer stellt Daten mit extra entwickelten Viren ein, die bei Ausführung einen ausgewählten Host kontaktieren, eine bestimme WEB-Seite aufrufen ...
  (2) Angreifer wird Mitglied im Netz und protokolliert Aktivitäten.
  Analyse $\mapsto$ Nutzerprofile. Abwehr: one-time reply-block, eigener Server.
- beim Server:
  (1) Angreifer erzeugt sehr große Teilstücke und prüft, wer sie aufnimmt (eintauscht) $\mapsto$ Kapazitätsanalyse, partielle Identifikation,
  (2) Abgleich mit Daten außerhalb des Systems (z.B. Bandbreite), Zuordnung Bandbreite - Netzknoten. Zuordnung zu IP-Netzverkehr zu Anwendung.
  (3) Trojan. Pherd berichtet, was ein FHP-Server hostet.
- beim Autor:
  Angreifer betreibt Server, protokolliert Einbringen von Dokumenten.
  Abwehr: Pseudonyme, one-time reply block.



Anonymisierung -Freenet

Idee bei Freenet:

- Zu jeder Datei erzeugt der Autor ein Kennwort K, wird mit Einweghash-Funktion verarbeit. Hashwert dient als ID für Datei. Das Paar (K,ID) wird veröffentlicht.
- Daten werden verschlüsselt, dabei wird K mit zur Schlüsselerzeugung verwendet.



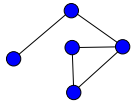
Anonymisierung (3) - Freenet - Details

- Mehrere Schlüsseltypen:

  - KSK (Keyword Signed Key)
  - SVK (Signature Verification Key)
  - SSK (SVK Subspace Key)
  - CHK (Content Hash Key)

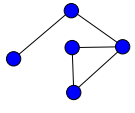
  Darstellung als Uniform Resource Identifiers (URI) freenet:TypSchlüsselwert

- KSK
  - Autor legt Kennwort K für Daten fest: DS/Vorl/P2P/Kap5
  - Aus K wird deterministisch ein P/Ö-ï£¡ erzeugt
  - Von ï£¡wird ein Hashwert H berechnet, Hashwert ist Dateikennung
  - Mit P werden Daten signiert
  - Daten werden mit K als Schlüssel verschlüsselt
- Suche: Sucher muß K kennen, welches der Autor verbreitet hat.
  H lokalisiert Daten, aber entschlüsselt sie nicht.



Freenet

- Autoren können Mengen aus mehreren Schlüssel verwalten
- Ähnlich KSK, aber zufällig generiert



Weitere Beispiele:

- Crowds: Nutzer bilden eine große Gruppe. Zufallsgesteuerte Weiterleitungsketten /Proxies.
  Latenzzeit?, kein Schutz für Autoren o. Serverbetreiber.
- FastTrack: kein Schutz der Anonymität.
- Usenet (Diskussionsforum): Server weltweit verteilt in verschiedenen Rechtsgebieten - Schutz vor Zensur.
- Publius: Schutz der Serverbetreiber, das Schlüssel zum Lesen (k, n) geteilt und verteilt gespeichert ist.