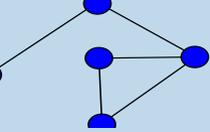

Vorlesung: P2P und Datenbanken (5)

Dr. Dieter Sosna

16. Juni 2008



Motivation

Identifikation / Authentifizierung

Gruppenzugehörigkeit

Nur Schlüsselaustausch

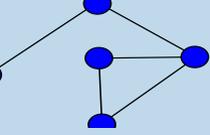
Authentifikation des Senders

Bestätigung des Empfangs

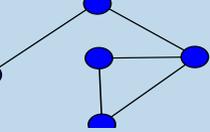
Geheimnisteilung

Anonymisierung

Zusammenfassung



Motivation

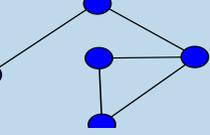


Warum?

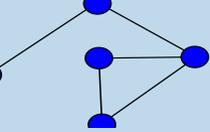
- Funktionalitätsklassen F7 - F9: Funktionen sicherer Systeme in Netz
 - ◆ Wechselseitige Identifikation und Authentifizierung
 - ◆ Unbestreitbarkeit der Sendung bzw. des Empfangs
 - ◆ Sichere Übertragung über unsicheren Kanal
 - ◆ Verbergen von Absender, Empfänger, Tatsache der Übertragung
- Sicherheit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

Lösungsansätze: Kryptographische Protokolle

Literatur: Bruce Schneier: Angewandte Kryptographie.
Protokolle, Algorithmen und Sourcecode in C.
Addison-Wesley, 1996 ISBN 3-89-319854-7



Identifikation / Authentifizierung



Begriffe:

- **Identifizierung:**

Nennung einer Identität gegenüber dem Partner.

- **Authentifizierung:**

Nachweis, dass der Teilnehmer die vorher von ihm genannte Identität tatsächlich besitzt.

Prüfung:

Existenzprüfung der Identität

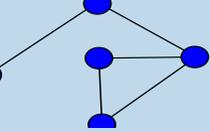
Verifizierung des Nachweises der Identität

Mitteilung eines negativen Ergebnisses als NEIN-Entscheidung (möglichst wenig Information über das System für den vermuteten Angreifer)

Authentifizierung durch:

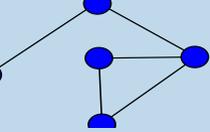
Wissen, Besitz, biometr. Daten (Sonderform des Besitzes)

Kein System ist unüberwindbar, Sicherheit, weil Kosten des Überwindens höher als der erwartete Nutzen.



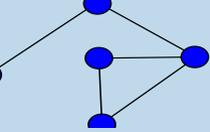
Identifikation / Authentifizierung (2)

- Wechselseitige Authentifizierung in reinem P2P-Systemen anfällig gegen *man-in-the-middle*-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 i.a. aus (Komplexität)
→ unsymmetr. Systeme oder Hybride Systeme (sym. Sitzungsschlüssel)



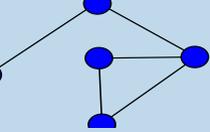
Identifikation mit Notar T

- **Annahme:** Notar macht keine Fehler, hat unbedingtes Vertrauen.
- Neuer Peer P weist auf sicherem Weg dem Notar seine Identität nach und **Notar gibt aus:** an den Peer dessen privaten Schlüssel, dessen öffentlichen Schlüssel verbunden mit dessen Identität und der Signatur des Notars, den öffentlichen Schlüssel des Notars.
Notar behält Kopie des öff. Schlüssels von P (mit Zuordnung zu P).
Notar vernichtet: alle Daten über den privaten Schlüssel von 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 P2P-System, Notar evt. Engpass,
Sicherheit des Systems = Sicherheit und Vertrauenswürdigkeit des Notar.



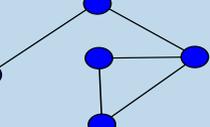
Identifikation mit Notar T : Echolink

- Kontakt mit Notar (Betreiber):
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. Behörden und weiteren Quellen. Positive Evaluierung \mapsto Peer erhält e-mail mit Zugangsdaten/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.
- Man-in-the-middle Angriff möglich, Angreifer muss Rückantwort mit Rufzeichen verbinden können (kein Problem, wenn eine WEB-Seite existiert).

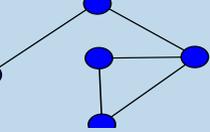


Identifikation ohne Notar

- Idealfall: 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.
- Real (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.
- Verfahren einseitig, Q hat keine Kontrolle, mit wem er kommuniziert (P2P-Netz erscheint wie ein Computer)
Symmetrische Variante: P hinterlegt bei Anmeldung bei Q, wie sich das System authentifiziert.
Q fragt dies beim Login ab: Q weiß, ob er im richtigen Netz ist.
Angriff, wenn P nicht korrekt arbeitet.
- Jeder unsichere Peer gefährdet das ganze Netz - Fehlererkennung ?



Gruppenzugehörigkeit



Anonyme Gruppenzugehörigkeit

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

- scheinbarer Widerspruch der Anforderungen

- Literatur:(Auswahl)

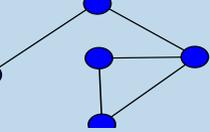
K.Koyama, „Demonstrating Membership of a Group Using the Shizuya-Koyama-Ithoh (SKI) Protokoll“, Proceedings of the 1989 Symposium on Cryptography and Information Security (SCIS 89), Gotenba, Japan, 1989

C. Shu, T. Matsumoto, and H. Imai, „A Multi-Purpose Proof System“, Transactions of 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 gemeinsames Geheimnis (das genau die Gruppenmitglieder kennen), dieses wird zum Nachweis der Mitgliedschaft benutzt.

Eintreten - Nachweis der Identität ?

Problem: Ausscheiden aus Gruppe - neues Geheimnis muß verteilt werden.



Gruppenzugehörigkeit mit Identifikation

Kommutative Einweg-Akkumulatoren

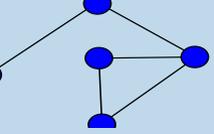
Literatur:

J.C. Benaloh and M. de Mare, „One-Way Accumulators: A Decentralized Alternative to Digital Signatures“, Advances in Cryptology - EUROCRYPT 93 Proceedings, Springer 1994, pp.274-285

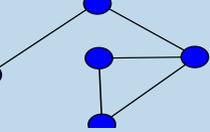
Daten werden nach (m,n) -Verfahren geteilt und auf Peers verteilt

- 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 hinzufügar, Streichung: neue Liste

Hinweis zu allen kryptographischen Verfahren: Verfahren gelten als bekannt!
Sicherheit schwindet mit der Zeit!



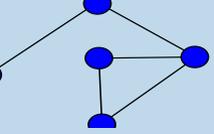
Nur Schlüsselaustausch



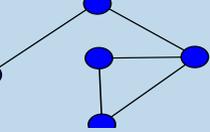
Schlüsselaustausch

in einem unsicheren Kanal

- - P, Q vereinbaren große Zahlen n, x (öffentlich)
 - P merkt sich p , Q q (geheim)
 - - P: $x_1 = x^p \pmod{n}$ an Q,
 - Q: $x_2 = x^q \pmod{n}$ an P
 - - P: $z_1 = x_2^p \pmod{n}$
 - Q: $z_2 = x_1^q \pmod{n}$
- P und Q kennen $z = z_1 = z_2$ (gemeinsames Geheimnis (n, z)).
- Das Berechnen des Logarithmus einer Zahl in einem Restklassenring gehört zu den schweren Problemen.
 - keine Authentifikation, kein Nachweis gegenüber einem Dritten möglich!
Umgekehrt kann das Auffinden von gleichen Paaren bei verschiedenen Peers auf Verbindung hindeuten.



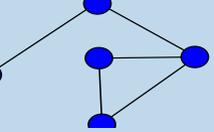
Authentifikation des Senders



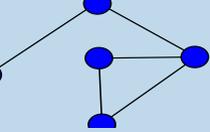
Authentifikation einer Nachricht

Nachweis, daß die Nachricht wirklich vom Absender stammt (Inhalt bleibt verdeckt).

- 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 Aussage über eine Kommunikation akzeptiert.
Problem: Sicherheit des Notars, Sicherung der Persistenz der Datenarchive. - Datenschutz?
- 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. Problem der Langzeitsicherheit.



Bestätigung des Empfangs

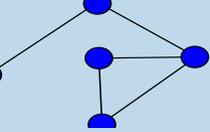


Bestätigung des Empfangs

Senden über Notar: kein Problem, Aussage des Notars gilt, Inhalt bleibt verdeckt.

Ohne Notar: Setzt Kooperation des Empfängers voraus.

- (Unsym. System:) 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 öffentlichen 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.

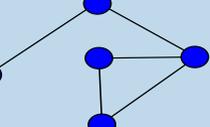
■ Oblivious Transfer (nicht eindeutige Übertragung)

Bruce Schneier: Angewandte Kryptographie, Add.Wesley, S.145

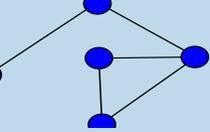
Die Nachricht wird mit einem symmetrischen Verfahren verschlüsselt, der Sitzungsschlüssel wird bitweise im Bestätigungsprotokoll übergeben und im Gegenzug wird die Bestätigung der Nachricht bitweise übertragen. (Eine Verquickung mit n Schlüsseln sorgt dafür, daß keine der beiden Parteien betrügen kann.)

■ Inhalt bzw. der Wert der Nachricht wird nicht gesichert.

■ Problem in P2P-Netzen: viele Nachrichten mit wenig Information

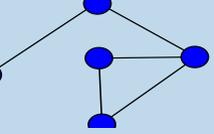


Geheimnisteilung

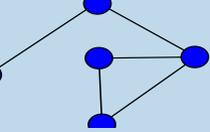


Geheimnisteilung

- Idee: Eine Nachricht m wird in N Teile zerlegt, nur alle Teile zusammen gestatten, die Nachricht zu rekonstruieren.
- M-N-Schwelwertverfahren: M von N Teilen reichen, die Nachricht zu rekonstruieren
- 2 Teile: s Zufallsstring, $t = m \text{ xor } s$
- M-N-Verfahren: m wird in M Teile zerlegt: $x_i, i=1, \dots, M$.
Diese dienen als Unbekannte in einem überbestimmten Gleichungssystem 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.

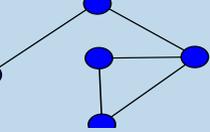


Anonymisierung



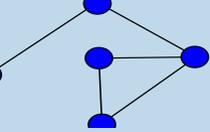
Anonymität

- 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 (2)

- 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)

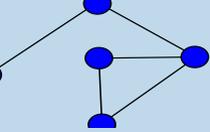
- - ◆ *Anonymität des Dokuments*

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.

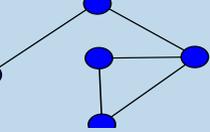
Punkte nicht unabhängig voneinander.



Anonymität der Übertragung

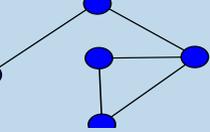
Ideen:

- Rundsprüche: unspezifizierte Adresse, jeder kann Nachricht empfangen.
(Wenn mit öS E des Empfängers verschlüsselt, erkennt nur der 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, \dots, K_n mit den Schlüsselpaaren $(E_i, D_i, i = 1 \dots n)$: A :
 $(E_1, (K_2, E_2(\dots(K_n, E_n(m)) \dots))) \mapsto K_1$, 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

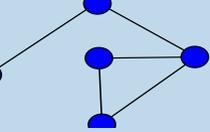
- Ein **Pseudonym** ist ein Attribut im Metadatensatz, dessen Werte jeweils genau einem Wert eines durch Anonymität verdeckten Attributs zugeordnet werden kann.
Beziehung: Pseudonym : Attribut = N:1
Das anonyme Attribut bleibt geschützt, aber alle Dokumente eines Pseudonyms als zusammengehörig erkennbar.
- 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

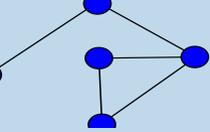
Literatur: Dingledine, R. u.a.: *The Free Haven Project: Distributed Anonymos Storage Service*. A.a.O.

- 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 $f \mapsto f_1, f_2, \dots, f_n$, (k, n) -Geheimnisteilung (k von n)
Dokumentspezif. Schlüsselpaar $(\text{ö}S_{doc}, pS_{doc})$ 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) , $\text{ö}S_{doc}$, *Signatur*))
Dokumentidentifikation Hashwert $H(\text{ö}S_{doc})$
- Suche:
Leser generiert $(\text{ö}S_{client}, pS_{client})$ und one-time remailer reply block.
Anfrage: $(H(\text{ö}S_{doc}), \text{ö}S_{client}, \text{reply} - \text{block})$ an alle dem Leser bekannten Serverknoten.



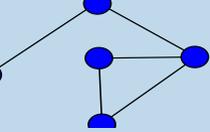
FHP: Anfrage bearbeiten, Gültigkeitszeit, Rückruf

- Jeder Server prüft, ob Daten zu $H(\text{ö}S_{doc})$ bei ihm vorhanden, ja: Verschlüsseln mit $\text{ö}S_{client}$ 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:

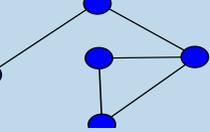
- Keine Garantie, daß R. vollständig gelingt (Knoten nicht erreichbar, Knoten verweigert Dienst oder Weiterleitung von x).
- $H(x)$ kann Urheber bloßstellen, wenn bei ihm x gefunden wird.
- Angreifer ermittelt, wer x besitzt und erpreßt Rückruf.

↳ *Rückruf in FHP nicht realisiert.*



FHP: Trading - Dokumentenaustausch

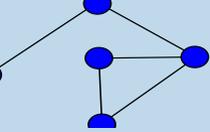
- Gründe:
 - Verbirgt Publikationstätigkeit u. Anfragen unter Tauschaktivität.
 - Erlaubt geordnetes Verlassen des Netzes, indem Server nur kurzlebige Dokumente akzeptiert.
 - Unterstützt lange Lebensdauer, es wäre sonst evt. schwierig, Server zu finden, die solche Dokumente akzeptieren.
 - Server kann 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 × 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

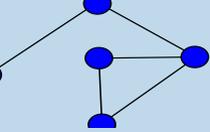
- beim Leser:
 - (1) Angreifer stellt Daten mit extra entwickelten Viren ein, die bei Ausführung einen ausgewählten Host kontaktieren, eine bestimmte WEB-Seite aufrufen ...
 - (2) Angreifer wird Mitglied im Netz und protokolliert Aktivitäten. Analyse \mapsto Nutzerprofile. Abwehr: one-time reply-block, 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. Pferd berichtet, was ein FHP-Server hostet.
- beim Autor:

Angreifer betreibt Server, protokolliert Einbringen von Dokumenten.
Abwehr: Pseudonyme, one-time reply block.



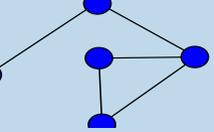
Idee bei Freenet:

- Zu jeder Datei erzeugt der Autor ein Kennwort K , wird mit Einweghash-Funktion verarbeitet. Hashwert dient als ID für Datei. Das Paar (K, ID) wird veröffentlicht.
- Daten werden verschlüsselt, dabei wird K mit zur Schlüsselerzeugung verwendet.
- Projekt „Darknet“ (2008)
 - sozial Netzwerke, Austausch mit Peers, zu denen Vertrauen besteht.



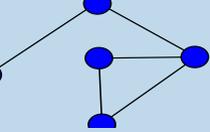
Anonymisierung - Freenet - Details

- 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-Paar erzeugt
 - ◆ Von Ö 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.



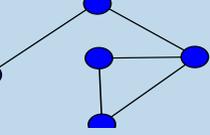
Freenet (4)

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

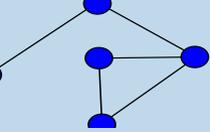


Weitere Beispiele:

- 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, da Schlüssel zum Lesen (k, n) geteilt und verteilt gespeichert ist.



Zusammenfassung



Zusammenfassung

- Schutzwürdigkeit von Sender, Empfänger, Inhalt und Tatsache der Kommunikation.
- Identifikation und Authentifizierung, Nachweis des Sendens und des Empfangs unabdingbar für kommerzielle Anwendungen.
- Protokolle einfach in hybriden P2P-SYstemen
- Beispiele für Grundprotokolle verschiedener Aufgaben.
Bisher keine kommerziellen Anwendungen bekannt.
Offen: Angriffserkennung - können Peers erkennen, wenn ein Peer nicht korrekt arbeitet.