

Mit Affen-Spielzeug etwas über Haustiere lernen

Evaluierung von PETs mittels APES-Bausteinen

Ulrich Flegel

Am Beispiel eines Audit-Daten-Pseudonymisierers wird gezeigt, wie APES-Anonymitätsbausteine zur informellen Entwurfsevaluierung gegebener datenschutzfördernder Systeme herangezogen werden können. Dabei ergeben sich Hinweise auf den Nutzen und die (Un-)Vollständigkeit des APES-Anonymitäts-Baukastens.¹

Einführung

Datenschutzfördernde Systeme, bzw. Privacy Enhancing Technologies (PETs), werden häufig im Hinblick auf spezifische Anwendungen entworfen. Dennoch beinhalten sie Funktionalität, die für andere Anwendungen wiederverwendbar ist. Im Projekt APES (Anonymity and Privacy in Electronic Services) [DCP03] definieren De Win et al. wiederverwendbare Anonymitäts-Bausteine [DWND+01], die folgendermaßen Verwendung finden können:

- 1) Ähnliche Bausteine lassen sich einfacher vergleichen als die wesentlich komplexeren Systeme, in denen sie verbaut sind.
- 2) Anhand einer Liste von Bausteinen und deren Eigenschaften lassen sich Unzulänglichkeiten in bestehenden Systemen systematisch identifizieren.
- 3) Datenschutzfördernde Systeme können unter Verwendung von Anonymitäts-Bausteinen systematisch synthetisiert werden.

Das APES-Projekt sowie der APES-Baukasten werden in den Abschnitten 1 und 2 kurz vorgestellt und in Abschnitt 4 verwendet, um den Entwurf eines gegebenen Pseudonymisierungs-Systems [BF00, Fle02] (s. Abschnitt 3) in Bausteine zu zerlegen. Dabei orientieren wir uns an den obigen Zielen und erarbeiten in Abschnitt 5 die folgenden Ergebnisse, die in Abschnitt 6 zusammengefasst werden:

- 1) Die im zerlegten Beispielsystem enthaltenen Bausteine werden mit Bausteinen ähnlicher Funktionalität verglichen, um festzustellen, ob der Entwurf durch den Austausch von Bausteinen mit stärkeren Eigenschaften optimiert werden kann.
- 2) Bei gegebenem Angreifer- und Vertrauensmodell des Beispielsystems können Unzulänglichkeiten im Entwurf informal identifiziert werden, indem der (ver-

meintlich) vollständige Baukasten von De Win et al. [DWND+01] als Entwurfsreferenz herangezogen wird.

- 3) Als Nebenergebnis erhalten wir Hinweise auf die (Un-)Vollständigkeit des APES-Baukastens hinsichtlich des evaluierten Entwurfs.

1 Projekt APES

Im Rahmen des APES-Projekts (Anonymity and Privacy in Electronic Services) wurde der Stand der Technik zu Privacy Enhancing Technologies erfasst und untersucht [SDDW+01]. Es wurden datenschutzfördernde Systeme für eine Palette von Anwendungen untersucht, u.a. anonyme Kommunikation, E-Mail, Web-Publishing, Web-Browsen, Zahlungsabwicklung, Wahlen sowie Auktionen. Für jede Anwendung wird dabei ein Überblick über die Funktionalität gegeben, es werden die beteiligten Parteien identifiziert, und es werden die Anforderungen und Eigenschaften der Anwendung beschrieben.

Für einige Anwendungen existieren verschiedene datenschutzfördernde Systeme, die Komponenten enthalten, die für spezifische Datenschutzaspekte in Form von Anonymität Verwendung finden. Im Rahmen des APES-Projekts wurden datenschutzfördernde Systeme in Anonymitäts-Bausteine zerlegt, die für andere Systeme wiederverwendet werden können. Der Fokus lag dabei auf *unbedingter Anonymität*, also nicht-aufhebbarer Anonymität. Die Anonymitäts-Bausteine wurden identifiziert, ihre Eigenschaften und Anforderungen beschrieben, sowie ihre Sicherheit und Korrektheit informal evaluiert [DWND+01]. Ferner wurde eine Ad-hoc-Methodik für die Synthese von datenschutzfördernden Systemen aus Bausteinen vorgeschlagen und für



Ulrich Flegel

Universität Dortmund, Fachbereich Informatik, Informationssysteme und Sicherheit, GI-FG-Leitung PET, SI-DAR (Vorsitz)

E-Mail: ulrich.flegel@udo.edu

¹ Die beschriebenen Arbeiten werden derzeit von der Deutschen Forschungsgemeinschaft gefördert unter Bi 311/10-3.

Der Titel dieses Beitrags spielt auf APES (engl. für Affen) = „Anonymity and Privacy in Electronic Services“ und PETs (engl. für Haustiere) = „Privacy Enhancing Technologies“ an.

den Entwurf von zwei Systemen exemplarisch verwendet [DNC+02].

2 Bausteine

Anonymitäts-Bausteine werden im APES-Projekt als spezifisch für die Verbindungsschicht oder Anwendungsschicht klassifiziert: Bausteine auf der *Verbindungsschicht* finden bei der Bereitstellung anonymer Kommunikationsverbindungen Verwendung, während Bausteine auf der *Anwendungsschicht* anwendungsspezifische Anonymitätsaspekte berücksichtigen. Für vollständige anonymitätswahrende Systeme werden Bausteine auf der Anwendungsschicht um Bausteine auf der Verbindungsschicht ergänzt.

Bausteine auf der Verbindungsschicht verbergen oder entfernen identifizierende Information, die auf dieser Schicht verfügbar sind. Identifizierende Information kann *explizit* auftreten, z.B. in der Form von IP-Adressen in IP-Paketköpfen. Eine Verbindung kann auch unter Verwendung *impliziter* Merkmale der Nachrichten-Erscheinungsform oder des Nachrichtenflusses entlang des Kommunikationspfades nachvollzogen werden. Nachrichten lassen sich anhand ihrer *Erscheinungsform* verketteten, etwa bzgl. Inhalt, Format oder Umfang. Auch der *Nachrichtenfluss* lässt sich über das Wissen um die Nachrichtenverarbeitung, z.B. Reihenfolge und Timing, rekonstruieren.

Zur Bereitstellung anonymer Kommunikationsverbindungen müssen sowohl explizit als auch implizit identifizierende Informationen verborgen werden. Entsprechend sind Bausteine zu kombinieren, welche die Erscheinungsform und den Nachrichtenfluss verändern (s. 2. Spalte in Tabelle 1).

Dafür werden in APES folgende Kompositionen vorgeschlagen:

seriell: Bausteine werden nacheinander ausgeführt, wobei die Ausgabe des zuvor ausgeführten Bausteins dem nächsten Baustein als Eingabe dient.

parallel: Funktional unabhängige Bausteine können nebenläufig ausgeführt werden, sofern höchstens einer der Bausteine die Erscheinungsform der Nachricht ändert.

verschachtelt: Die Ausführung des äußeren Bausteins wird für die Ausführung des inneren Bausteins ausgesetzt.

Bausteine auf der Anwendungsschicht verbergen oder entfernen identifizierende Informationen, die auf dieser Schicht verfügbar sind. Sie implementieren Techniken,

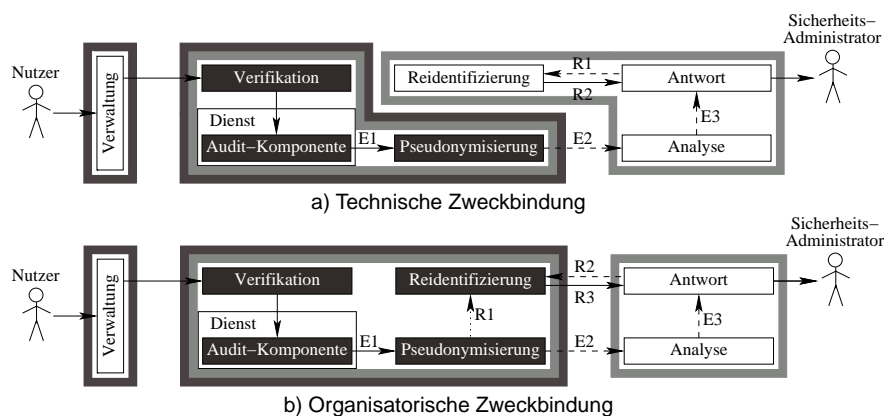


Abbildung 1: Architekturmodell des Pseudonymisierer-Beispielsystems

die entwickelt wurden, um Anonymität in einer spezifischen Anwendung zu erreichen (s. 3. Spalte in Tabelle 1).

3 Beispiel Pseudonymisierung

Im Folgenden wird ein Beispielsystem vorgestellt, anhand dessen die Vorgehensweise bei der Evaluation von PETs erläutert wird.

Moderne Mehrbenutzer-Dienste und -Betriebssysteme stellen z.B. für Zwecke der Sicherheit und Abrechnung Audit-Daten (Log-Daten) zur Verfügung. Diese lassen sich im Normalfall ohne nennenswerten Aufwand in Bezug zu den Nutzern, also natürlichen Personen, setzen. Entsprechend der Datenschutz-Gesetzgebung und der Nutzererwartung an Anonymität ist die Aufzeichnung von Audit-Daten problembehaftet. Um dieses Problem zu vermeiden, können Audit-Daten vor ihrer Speicherung so pseudonymisiert werden, dass einerseits dem Datenschutz Rechnung getragen wird und andererseits im Einzelfall und zweckgebunden Zurechenbarkeit herstellbar ist.

Wir schlagen ein entsprechendes Pseudonymisierungssystem vor [BF00] und stellen für die Pseudonymisierung von Unix-Audit-Daten eine Implementierung namens *Pseudo/CoRe* zur Verfügung [Fle02]. Bei korrektem Einsatz können während des Normalbetriebs lediglich mit *Pseudo/CoRe* pseudonymisierte Audit-Daten auf Missbrauchssymptome analysiert werden. Nur bei hinreichendem Missbrauchsverdacht (Zweckbindung) können die personenbeziehbaren Daten unmittelbar wiederhergestellt werden, um Zurechenbarkeit im Einzelfall zu gewährleisten.

Im Vergleich zu anderen Lösungen bietet dieser Ansatz diverse Vorteile, z.B. eine technische Zweckbindung, die Möglichkeit zur unmittelbaren Pseudonym-Aufdeckung - also unabhängig von Dritten, sowie einen geringen Aufwand bei der Einführung, da der Ansatz unabhängig von Endsystemen und aufwändigen (PKI-)Infrastrukturen arbeitet [Fle03b].

Abb. 1 illustriert in einem Architekturmodell den Fluss identifizierender (durchgezogene Pfeile) und pseudonymisierter (gestrichelte Pfeile) Daten, sowie den Fluss der Pseudonym-Zuordnungsregel (gepunkteter Pfeil), wenn ein Nutzer auf einen Dienst zugreift, der Audit-Daten erzeugt und unmittelbar danach pseudonymisiert.

Der Nutzer kontrolliert im Modell in Abb. 1 die ausschließlich dunkelgrau gerahmten Komponenten, während der Sicherheitsadministrator (SA) die ausschließlich hellgrau gerahmten Komponenten kontrolliert. Doppelt gerahmte Komponenten werden vom Datenschutzbeauftragter (DB) der Organisation kontrolliert. Diese Komponenten sind im Sinne multilateraler Sicherheit vom DB so zu konfigurieren, dass sie einen fairen Ausgleich zwischen dem individuellen Nutzerinteresse an Anonymität einerseits und dem Interesse an Zurechenbarkeit im Missbrauchsfall andererseits durchsetzen.

Für Details zum Vertrauensmodell, dem Prozess der Pseudonymisierung, sowie der technischen und organisatorischen Zweckbindung bei der kontrollierten Pseudonym-Aufdeckung sei auf die Arbeiten von Flegel verwiesen [Fle03a, Fle03b].

Für die Pseudonymisierung von Audit-Daten gelten spezifische Anforderungen, die auch das Beispielsystem zu erfüllen hat.

Der Pseudonymisierer muss in der Lage sein, das vom Dienst erzeugte Audit-Daten-

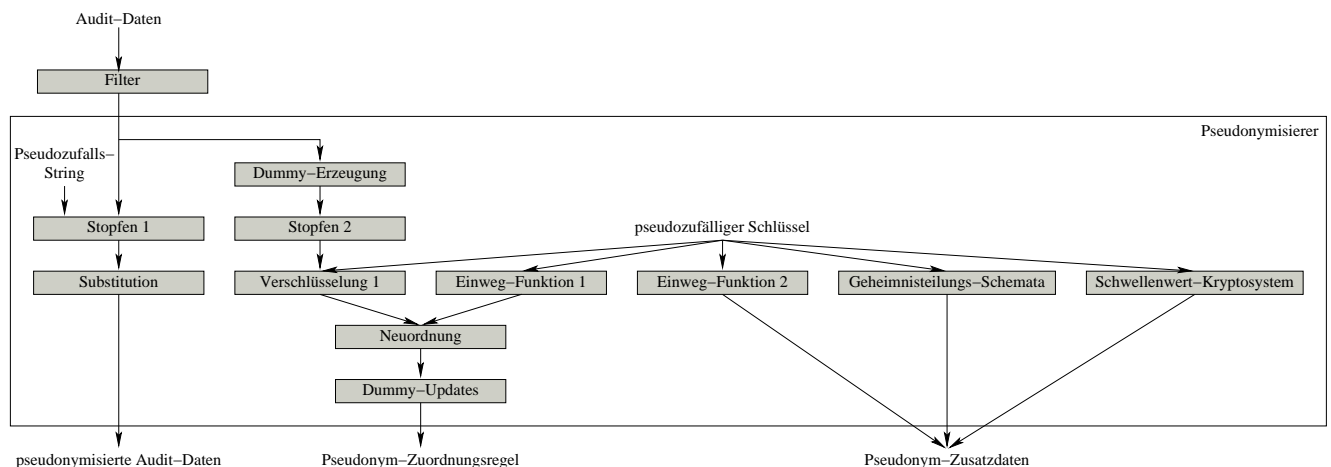


Abbildung 2: Zerlegung des Beispielsystems in Bausteine

Volumen *on-the-fly* zu bewältigen. Dementsprechend können keine Anonymitäts-Bausteine zur Anwendung kommen, die eine hohe Berechnungskomplexität aufweisen und/oder signifikante Latenzen erzeugen [Fle03b]. Leistungsmessungen haben bestätigt, dass das implementierte Beispielsystem *Pseudo/CoRe* diesen Anforderungen gerecht wird [Fle02].

4 Zerlegung

Das Beispielsystem wurde unter der Annahme bzw. Voraussetzung entworfen, dass die SAs das Nutzerverhalten lediglich durch Inspektion pseudonymisierter Audit-Daten beobachten können. Dies impliziert, dass die SAs nicht die Zugriffe der Nutzer über das Netz beobachten können, sondern lediglich sehen können, von wo die pseudonymisierten Audit-Daten stammen. Da die letztere Information nicht schützenswert ist, sind mit den genannten Voraussetzungen keine Maßnahmen auf der Verbindungsschicht notwendig.

Falls die SAs doch Nutzerzugriffe über das Netz beobachten können, sind sie in der Lage, die Zugriffsbeobachtungen mit den pseudonymen Audit-Daten zu korrelieren. In diesem Fall ist die Notwendigkeit für Anonymität auf der Verbindungsschicht gegeben. Entsprechende Maßnahmen können dann unabhängig vom Beispielsystem mittels existierender Lösungen für anonyme Verbindungen getroffen werden (s. Seys et al. [SDDW+01]).

Abb. 2 zeigt eine Zerlegung des konzeptionellen Entwurfs des Beispielsystems gemäß Flegel und Biskup [BF00]. Der *Filter*-Baustein dient der Reduktion des Audit-Datenstroms auf das für die Miss-

brauchserkennung notwendige Maß. Der *Pseudonymisierer* durchsucht jeden eingehenden Audit-Datensatz nach vorab vom DB definierten, zu verbergenden personenbeziehbaren Merkmalen. Jedes dieser Merkmale wird durch einen pseudozufälligen String (Pseudonym) ersetzt, der optional auf eine definierte Länge angepasst/gestopft ist und den Anforderungen der Audit-Daten-Analyse-Werkzeuge hinsichtlich Merkmals-Verkettbarkeit genügt (s. *Pseudozufalls-String* und *Stopfen 1* und *Substitution* in Abb. 2).

Für jeden zu pseudonymisierenden Merkmalstyp definiert der DB vorab die Möglichkeit zur kontrollierten Pseudonym-aufdeckung. Bei gewünschter Aufdeckungsmöglichkeit eines Pseudonyms wird das entsprechende Klartextmerkmal verschlüsselt (s. *Verschlüsselung 1* und *pseudozufälliger Schlüssel* in Abb. 2). Folglich ist für die Pseudonymaufdeckung die Rekonstruktion des korrekten Schlüssels erforderlich. Die Pseudonym-Zuordnungsregel besteht aus Paaren von Merkmalskryptogrammen und Schlüssel-Digests, wobei letztere als Suchkriterium für rekonstruierte Schlüssel dienen (s. *Verschlüsselung 1*, *Einweg-Funktion 1* und *Pseudonym-Zuordnungsregel* in Abb. 2).

Zur Vergrößerung der Systemteilnehmeranzahl werden Dummy-Einträge in der Pseudonym-Zuordnungsregel erzeugt. Um die Verkettbarkeit der Einträge zu verhindern, werden diese auf eine gemeinsame Länge angepasst/gestopft und zufällig neu geordnet (s. *Dummy-Erzeugung*, *Stopfen 2* und *Neuordnung* in Abb. 2). Um Inferenzen mittels der Anzahl der Updates der Pseudonym-Zuordnungsregel zu erschweren, werden Dummy-Updates durchgeführt,

wenn keine Einträge eingefügt wurden [BF00] (s. *Dummy-Updates* in Abb. 2).

Die *Pseudonym-Zusatzdaten* enthalten u.a. Informationen für die kontrollierte Pseudonym-Aufdeckung. Diese Zusatzdaten werden in speziellen Audit-Datensätzen in den Audit-Datenstrom eingebettet. Voraussetzung für die Nutzung der Pseudonym-Zuordnungsregel für die kontrollierte Pseudonym-Aufdeckung ist ein gültiger Schlüssel. Ein solcher kann unter Einhaltung von organisatorischer oder technischer Zweckbindung aus den Pseudonym-Zusatzdaten gewonnen werden. Die organisatorische Zweckbindung wird durchgesetzt, indem der notwendige Schlüssel mittels eines *Schwellenwert-Kryptosystems* so verschlüsselt wird, dass er nur von a priori definierten Personengruppen kooperativ entschlüsselt werden kann.

Die technische Zweckbindung wird mittels kryptographischer *Geheimnisteilungs-Schemata* durchgesetzt, indem der notwendige Schlüssel in Anteile aufgeteilt wird, die sukzessive in den Pseudonym-Zusatzdaten verfügbar gemacht werden. Missbrauchsszenarien werden dabei primär vermöge geeigneter Schwellenwerte für Shamir Geheimnisteilungs-Schemata modelliert sowie durch die Zuordnung von Missbrauchskontexten und Gewichten zu Pseudonymen in Beobachtungen potentiell missbrauchsbezogener Aktivitäten [BF00].

Ein gültiger Schlüssel kann dann aus den Anteilen zu Pseudonymen rekonstruiert werden, die Missbrauchskontexten zugeordnet sind, wenn ein Missbrauchsverdacht hinreichend erfüllt ist, d.h. der Schwellenwert eines Geheimnisteilungs-Schemas überschritten ist. Dementsprechend ist die kontrollierte Pseudonym-Aufdeckung tech-

Tabelle 1: Anonymitäts-Bausteine, die im Beispielsystem Verwendung fanden; „Baustein“ = bedingte Anonymität, „!“ = fehlende Klassifizierung, „ “ = fehlender Baustein, „?“ = Kandidat für die Optimierung des Beispielsystems

Baustein	Verbindungsschicht		Anwendungsschicht	Beispielsystem
	Erscheinungsform	Nachrichtenfluss		
Verschlüsselung	√		√	√
Stopfen	√		!	√
Substitution	√		!	√
Kompression	√			
Neuordnung		√	!	√
Latenz		√		?
Dummy-Aktivität		√	!	√
Keine Wiedereinspielung		√		
Filter		√	!	√
Cache		√		
Broadcast		√	√	
Multiplexen		√		
Schwarzes Brett		√	√	
Einweg-Funktion	-	-	-	√
(faire) blinde Signatur			√	?
Gruppensignatur			√	?
Schwellenwert-Kryptosystem			√	√
Mehrparteien-Berechnung			√	?
Homomorphe Verschlüsselung			√	?
Abstreitbare Verschlüsselung			√	
Geheimnisteilungs-Schemata			√	√
Zero-Knowledge			√	?
Pseudonyme			√	? / √
Vertrauenswürdige Dritte			√	√

nisch an Zwecke gebunden, welche die Zurechenbarkeit hinreichender Missbrauchsverdachte erfordern.

Da die beschriebenen Anteile per se unverkettbar sind, können aus beliebigen Anteilskombinationen auch ungültige Schlüssel rekonstruiert werden. Tritt dieser Fall ein, lässt sich der Digest des rekonstruierten Schlüssels in der Regel nicht mit den Schlüssel-Digests in der Pseudonym-Zuordnungsregel abgleichen (s. *Einweg-Funktion 1* in Abb. 2). Um Fehltreffer entdecken zu können, also gültige Schlüssel, die jedoch aus Schlüssel-Anteilen verschiedener Merkmale rekonstruiert wurden, ist jeder Anteil mit einem weiteren Digest ausgestattet, der die Zuordnung des Anteils zum ursprünglichen Schlüssel abbildet [BF00] (s. *Einweg-Funktion 2* in Abb. 2).

Da das Suchen gültiger Kombinationen unverkettbarer Anteile eine unpraktikable Berechnungskomplexität aufweist, wurden für die Implementierung des Systems Vereinfachungen vorgenommen. Für die Zerlegung des vereinfachten und tatsächlich implementierten Entwurfs des Beispielsystems sei auf die Arbeiten von Flegel ver-

wiesen [Fle02, Fle05]. Zum Pseudonymisierer wurde ein passender Reidentifizierer für die Pseudonym-Aufdeckung entworfen. Dieser lässt sich ebenfalls in Bausteine zerlegen, wobei allerdings keine weiteren Einsichten gewonnen wurden. Die Details werden daher hier nicht dargestellt.

5 Analyse

In der 4. Spalte von Tabelle 1 sind die Bausteine zusammengefasst, die bei der Zerlegung des gegebenen Beispielsystems in Abschnitt 4 identifiziert wurden. Obwohl das Beispielsystem lediglich Anonymität in der Anwendungsschicht herstellen soll, besteht es aus Verbindungsschicht-Bausteinen, die im APES-Projekt nicht für die Anwendungsschicht vorgesehen waren (s. „!“ in der 3. Spalte in Tabelle 1).

Die APES-Bausteine zielen primär auf die Bereitstellung unbedingter Anonymität. Ausnahmen davon wurden in APES gekennzeichnet und entsprechend in der 1. Spalte in Tabelle 1 hervorgehoben. Das Beispielsystem zielt allerdings auf bedingte, also aufhebbare Anonymität. Bausteine für

bedingte Anonymität, die das Beispielsystem nicht verwendet, können für eine Optimierung des Systems in Betracht gezogen werden. Es sei angemerkt, dass in dem primär für unbedingte Anonymität ausgelegten APES-Baukasten bereits alle Bausteine definiert sind, die das Beispielsystem für bedingte Anonymität verwendet.

Man beachte, dass das Beispielsystem einen elementaren Baustein namens *Einweg-Funktion* verwendet, der es erlaubt, verborgene Merkmale auf Identität zu vergleichen. Eine kollisionsresistente Einweg-Funktion h , im Beispielsystem eine kryptographische Hashfunktion, wird verwendet, um Merkmale dergestalt zu verbergen, dass für zwei gegebene Merkmale f_i und f_j bei $h(f_i) = h(f_j)$ mit hoher Wahrscheinlichkeit $f_i = f_j$ gilt.

Anhand der durch Zerlegung gewonnenen Baustein-Liste wie in Tabelle 1 und mit der Dokumentation aus dem APES-Projekt lassen sich ggf. nicht verwendete Bausteine mit ähnlicher Funktionalität und stärkeren Eigenschaften identifizieren. So lassen sich Kandidaten für eine Optimierung des Entwurfs identifizieren bzw. ungeeignete Bausteine ausschließen (s. „?“ in der 4. Spalte in Tabelle 1).

Die Liste hilft auch Mängel zu entdecken, z.B. indem die dem untersuchten System zugrundeliegenden Annahmen Maßnahmen für eine Schicht einfordern, die im Systementwurf fehlen. Ebenfalls wäre es möglich, dass im Systementwurf auf der Verbindungsschicht Vorkehrungen für die Veränderung des Erscheinungsbildes oder des Nachrichtenflusses fehlen, so dass ein Angreifer identifizierende Information erlangen kann.

Für das Beispielsystem wurde diese Vorgehensweise von Flegel detailliert beschrieben [Fle05]. Bei der Abarbeitung der Liste entsteht eine Dokumentation, die Entwurfsentscheidungen rechtfertigt, bzw. die Auswahl oder den Verwurf von alternativen Bausteinen begründet. Durch diese systematische Vorgehensweise und unter Rückgriff auf Fremdwissen, also den Baukasten, wird der Entwurf hinsichtlich der Anonymitätseigenschaften transparent und es kann die Gewissheit erhöht werden, dass nach dem aktuellen Stand der Technik geeignete Bausteine gewählt und offensichtliche Mängel vermieden wurden.

Fazit

Ein Beispielsystem zur Pseudonymisierung von Audit-Daten wurde in Anonymitäts-Bausteine zerlegt und diese mit den im APES-Projekt definierten Bausteinen abgeglichen, um auf informale Weise

- 1) Unzulänglichkeiten und
- 2) Potential für Entwurfsverbesserungen zu identifizieren.

Im Hinblick auf Ziel 1 konnten keine Mängel im Entwurf des Beispielsystems festgestellt werden. Zu Zielstellung 2 konnten für das Beispielsystem sieben Kandidaten-Bausteine identifiziert werden, um z.B. die Machtstellung des Vertrauenswürdigen Dritten aufzuweichen und zusätzliche Pseudonymeigenschaften zu erhalten. Allerdings erfüllen die Kandidaten-Bausteine nicht die speziellen Anforderungen des Beispielsystems hinsichtlich Berechnungskomplexität und Latenzen (vgl. Abschnitt 3). Demnach lässt sich der Entwurf des Beispielsystems nur verbessern, wenn diese Anforderungen aufgehoben werden, sich also Bausteine mit stärkeren Eigenschaften gegen Zeit- und Berechnungsanforderungen aufwiegen lassen. Der Entwurf des Beispielsystems wurde nicht verändert, in der nun größeren Gewissheit, dass er unter den gegebenen Annahmen und Anforderungen geeignet ist.

Bei der exemplarischen Systemzerlegung in Bausteine wurden Hinweise auf Eigenschaften des APES-Ansatzes gewonnen. Es ist grundsätzlich möglich, Entwürfe von datenschutzfördernde Systemen mit Hilfe des APES-Baukastens zu untersuchen, um Schwächen und Optimierungspotentiale zu identifizieren.

Die Stärke der dabei gewonnenen Aussagen hängt von der Vollständigkeit des APES-Baukastens ab. Obwohl De Win et al. [DWND+01] einen vollständigen Baukasten angeben wollten, scheint es unwahrscheinlich, dass künftige Entwicklungen datenschutzfördernder Technologien dem Baukasten keine neuen Bausteine hinzufügen werden.

Tatsächlich war es im Beispielsystem notwendig, verborgene Merkmale zu vergleichen, was mit Hilfe von Einweg-Funktionen erreichbar ist. Sicherlich sind Einweg-Funktionen bereits Bestandteil diverser APES-Bausteine. Dennoch sind sie als eigenständige Bausteine für viele PET-Anwendungen interessant. Daher postulieren wir einen entsprechenden Baustein *Einweg-Funktion* für die Anwendungsschicht. Dies weist darauf hin, dass der APES-Baukasten bereits zum damaligen Stand der Technik unvollständig war.

Für die Evaluierung des Beispielsystems gilt also, dass bessere Entwürfe und unentdeckte Mängel nicht ausgeschlossen werden können. Die qualitativen Aussagen über den Entwurf sind daher lediglich als starke Indikatoren zu verstehen, die auf dem gegenwärtigen Wissensstand beruhen.

Zusätzlich zur Unvollständigkeit der Liste der im APES-Baukasten definierten Bausteine ließ sich feststellen, dass die Schichtenklassifizierung von mindestens fünf Bausteinen unvollständig ist, da sie im Beispielsystem auf der Anwendungsschicht eingesetzt werden, obwohl sie in APES nur für die Verbindungsschicht vorgesehen waren. Weitere Untersuchungen könnten ergeben, dass nahezu alle Verbindungsschicht-Bausteine auch auf der Anwendungsschicht nutzbringend einsetzbar sind.

Schließlich sei noch angemerkt, dass die im APES-Baukasten vorhandenen Bausteine bereits ausreichen, um das Beispielsystem mit bedingter, also aufhebbarer Anonymität auszustatten, obwohl im APES-Projekt zunächst primär die unbedingte Anonymität in den Blick genommen wurde. Es könnte sich herausstellen, dass der dafür eigentlich nicht ausgelegte APES-Baukasten bereits ausreicht, um viele PETs mit bedingter Anonymität zu synthetisieren.

Literatur

- [BF00] J. Biskup, U. Flegel. Threshold-based Identity Recovery for Privacy Enhanced Applications. In S. Jajodia et al. (Hrsg.), 7th ACM Conference on Computer and Communications Security (CCS 2000), S. 71–79, Athen, November 2000.
- [DCP03] C. Díaz, J. Claessens, B. Preneel. APES — Anonymity and Privacy in Electronic Services. Datenschutz und Datensicherheit (DuD), 27(3):143–145, März 2003.
- [DNC+02] C. Díaz, V. Naessens, J. Claessens, B. De Win, S. Seys, B. De Decker, B. Preneel. Anonymity and Privacy in Electronic Services (APES) Deliverable 5 – Tools for Technologies and Applications. Bericht, K. U. Leuven, November 2002.
- [DWND+01] B. De Win, V. Naessens, C. Díaz, S. Seys, C. Goemans, J. Claessens, B. De Decker, J. Dumortier, B. Preneel. Anonymity and Privacy in Electronic Services (APES) Deliverable 3 – Technologies Overview. Bericht, K. U. Leuven, November 2001.
- [Fle02] U. Flegel. Pseudonymizing Unix Log Files. In G. Davida et al. (Hrsg.), Infrastructure Security Conference (InfraSec2002), LNCS 2437, S. 162–179, Bristol, Oktober 2002. Springer.
- [Fle03a] U. Flegel. Anonyme Audit-Daten im Überblick. Datenschutz und Datensicherheit (DuD), 27(5):278–281, Mai 2003.
- [Fle03b] U. Flegel. Ein Architektur-Modell für anonyme Autorisierungen und Überwachungsdaten. In R. Grimm et al. (Hrsg.), 1. GI Konferenz Sicherheit – Schutz und Zuverlässigkeit (Sicherheit 2003), LNI P-36, S. 293–304, Frankfurt, September 2003. Köllen.
- [Fle05] U. Flegel. Evaluating the Design of an Audit Data Pseudonymizer Using Basic Building Blocks for Anonymity. In H. Federrath (Hrsg.), 2. GI Konferenz Sicherheit – Schutz und Zuverlässigkeit (Sicherheit 2005), LNI P-62, S. 221–232, Regensburg, April 2005. Köllen.
- [SDDW+01] S. Seys, C. Díaz, B. De Win, V. Naessens, C. Goemans, J. Claessens, W. Moreau, B. De Decker, J. Dumortier, B. Preneel. Anonymity and Privacy in Electronic Services (APES) Deliverable 2 – Requirement Study of Different Applications. Bericht, K. U. Leuven, Mai 2001.
- [BF00] J. Biskup, U. Flegel. Threshold-based Identity Recovery for Privacy Enhanced Applications. In S. Jajodia et al. (Hrsg.), 7th ACM Conference on Computer and