Langfristige Sicherheit am Beispiel eines virtuellen Tresors

Lucie Langer, Alex Wiesmaier

Technische Universität Darmstadt
Kryptographie und Computeralgebra
[langer|wiesmaier]@cdc.informatik.tu-darmstadt.de

Ab November 2010 wird in Deutschland der elektronische Personalausweis an die Bürger ausgegeben. Ziel dieser Einführung ist es, den bisherigen Personalausweis um elektronische Funktionen zu ergänzen und damit den Herausforderungen und Möglichkeiten der Gegenwart anzupassen [BdI08]. Neben einer zusätzlichen Aufnahme biometrischer Merkmale für hoheitliche Personenkontrollen wird der Ausweis seinem Halter einen elektronischen Identitätsnachweis über das Internet ermöglichen. Darüber hinaus wird der elektronische Personalausweis auch die Option qualifizierter elektronischer Signaturen bieten. Dadurch werden zahlreiche neue Anwendungen in eGovernment und eBusiness möglich gemacht, die dem Bürger ein erhöhtes Maß an Flexibilität und Mobilität bieten.

Eine innovative Anwendung des elektronischen Personalausweises ist Thema eines interdisziplinären Forschungsprojekts, dessen Vorhaben im Folgenden beschrieben wird. Ziel des Projekts ist die Entwicklung eines Konzepts zur langfristig sicheren Aufbewahrung und mobilen Bereitstellung persönlicher Daten durch ein elektronisches Schließfach: Der *Lifetime eSafe* erlaubt es dem Nutzer, wichtige persönliche Dokumente langfristig sicher und vertraulich abzulegen und ermöglicht gleichzeitig den mobilen Zugriff auf diese Dokumente. Des Weiteren soll der Benutzer des eSafe auch anderen Personen Zugriff auf ausgewählte Daten gewähren können. Der elektronische Personalausweis ermöglicht die sichere Authentisierung des Benutzers am eSafe. Für den Prototypen des eSafe sind im Einzelnen folgende Funktionalitäten eingeplant:

- Die Nutzer können beliebige Daten sicher auf ihren eSafe hochladen. Typischerweise sind dies auf Lebenszeit gültige Dokumente wie VBL-Bescheide, Zeugnisse, Empfehlungsschreiben, usw.

- Die Nutzer können feingranular (z.B. einzelne Dokumente, einzelne Personen) und grobgranular (z.B. Dokumentengruppen, Personengruppen) Zugriffsrechte für von den Nutzern ausgewählte Teile des eSafes geben. Diese Rechte können zeitlich limitiert werden, und es besteht außerdem die Möglichkeit, die Anzahl der Zugriffe zu beschränken. Eine Beispielanwendung hierfür sind Zeugnisse und Empfehlungsschreiben, die im Rahmen von Bewerbungen den Entscheidern zur Verfügung gestellt werden; durch den eSafe wird hier das Verschicken umfangreicher eMail-Anhänge vermieden.

Die Vertraulichkeit der im eSafe gespeicherten Daten wird dadurch gewährleistet, dass der eSafe von einem verteilten Dienstleisterkonsortium betrieben wird. Selbst wenn mehrere Mitglieder des Konsortiums miteinander kooperieren, um die Vertraulichkeit der Daten zu kompromittieren, bleiben die Daten geheim, solange ein konfigurierbarer Anteil der Konsortialpartner nicht kollaboriert. Dies wird garantiert durch den Einsatz eines Speicherkonzepts, welches auf Shamirs Secret Sharing [Sha79] zurückgeht und bereits von Doi et al. beschrieben wurde

[MDNK08]. Die grundlegende Idee besteht darin, dass eine Datei auf eine bestimmte Anzahl von Blöcken aufgeteilt wird. Jeder dieser Blöcke wird dann als eigenständiges Geheimnis betrachtet und in Form von *Shares* auf die n Konsortialpartner verteilt. Nach dem Muster des Secret Sharing mit den Parametern (n, k) [Sha79] werden dann zur Bereitstellung einer Datei k verschiedene Shares der n Konsortialpartner benötigt. Weniger als k Konsortialpartner sind dagegen nicht in der Lage, eine Datei zu rekonstruieren. Gleichzeitig hat man hierdurch ein skalierbares Maß an Redundanz: Der Verlust oder die Kompromittierung von bis zu $n-k$ Shares ist unkritisch. Der genaue Ablauf der verteilten Speicherung und Rekonstruktion wird weiter unten näher beschrieben.

Komponenten

Der eSafe beinhaltet im Wesenlichen die folgenden Komponenten:

- Client: PC, von dem aus der Nutzer agiert.
- Webserver: Stellt dem Client die Seite mit den benötigten Skripten zur Verfügung, über die der Nutzer auf seine Daten zugreifen kann. Der Webserver ist aber nicht im Besitz der eigentlichen Dateien, sondern enthält nur eine Datenbank mit deren Metainformationen, die er dem Client präsentiert.
- Storageserver: Auf den Storageservern werden die Shares gespeichert. Dabei soll jeder Konsortialpartner einen Storageserver betreiben und sich um die ordnungsgemäße Wartung sowie um ein regelmäßiges Backup kümmern.
- Authentifizierungsapplet: Wird benutzt, um sich mit Hilfe des elektronischen Personalausweises gegenüber dem Webserver zu authentifizieren.
- Übertragungsapplet: Generiert die Shares auf dem Client und überträgt sie an die Storageserver bzw. holt die Shares von den Storageservern und setzt diese wieder zusammen.

Generieren der Shares

Wir definieren zunächst kurz die Vorgehensweise zur Verteilung eines Geheimnisses nach [Sha79]: Sei F ein algebraischer Körper, $S \in F$ das Geheimnis, $k, n \in \mathbb{N}$ mit $k \leq n$. Wähle $k-1$ geheime Koeffizienten $a_i \in F, i = 1, 2, \ldots, k-1$ des Polynoms

$$f(x) = \sum_{i=0}^{k-1} a_i x^i = S + \sum_{i=1}^{k-1} a_i x^i$$

mit $a_0 = S$, d.h. das Geheimnis ist der konstante Term des Polynoms. Wähle n paarweise verschiedene IDs $x_i \in F, i = 1, 2, \ldots, n$. Für die n Paare $(x_i, f(x_i)) = (x_i, y_i)$ mit $i = 1, 2, \ldots, n$ gilt

$$\begin{pmatrix} y_1 \\ y_2 \\ \\ \vdots \\ \\ y_n \end{pmatrix} = \underbrace{\begin{pmatrix} 1 & x_1 & x_1^2 & \dots & x_1^{k-1} \\ 1 & x_2 & x_2^2 & \dots & x_2^{k-1} \\ & & & & \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ & & & & \\ 1 & x_n & x_n^2 & \dots & x_n^{k-1} \end{pmatrix}}_{n \times k \text{ Matrix}} \begin{pmatrix} S \\ a_1 \\ a_2 \\ \vdots \\ \\ a_{k-1} \end{pmatrix}$$

Dabei ist y_i der Geheimnisteil zur ID x_i. Die y_i werden nun an die n verschiedenen Instanzen verteilt.

Um eine Datei im eSafe zu speichern, wird diese zunächst in b Blöcke der Länge m Bits aufgeteilt. Jeder Block wird einzeln als Geheimnis S betrachtet, und mit obiger Gleichung berechnet man in b Iterationen für feststehende IDs die zugehörigen Geheimnisteile. Geheimnisteile mit selber ID werden daraufhin in einer Datei zusammengefasst und bilden ein sogenanntes **Share**. Insgesamt gibt es also n Shares (bestehend aus je b Geheimnisteilen), die auf den n verschiedenen Storageservern gespeichert werden.

Um eine Datei zu rekonstruieren, muss jeder der b Blöcke rekonstruiert werden, aus denen die Datei besteht. Jeder dieser Blöcke bildet ein Geheimnis S. Um S zu rekonstruieren benötigt man k Paare $(x_{j_i}, y_{j_i}), i = 1, 2, \dots, k$ aus ID und zugehörigem Geheimnisteil:

$$\begin{pmatrix} S \\ a_1 \\ a_2 \\ \vdots \\ \\ a_{k-1} \end{pmatrix} = \underbrace{\begin{pmatrix} 1 & x_{j_1} & x_{j_1}^2 & \dots & x_{j_1}^{k-1} \\ 1 & x_{j_2} & x_{j_2}^2 & \dots & x_{j_2}^{k-1} \\ & & & & \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ & & & & \\ 1 & x_{j_k} & x_{j_k}^2 & \dots & x_{j_k}^{k-1} \end{pmatrix}^{-1}}_{k \times k \text{ Matrix}} \begin{pmatrix} y_{j_1} \\ y_{j_2} \\ \\ \vdots \\ \\ y_{j_k} \end{pmatrix}$$

Das Geheimnis kann nun durch Lagrange-Interpolation rekonstruiert werden:

$$S = f(0) = \sum_{i=1}^{k} y_{j_i} \prod_{l=1, l \neq i}^{k} \frac{x_{j_l}}{x_{j_l} - x_{j_i}}$$

Auf diese Weise wird über die Rekonstruktion der einzelnen Blöcke die gesamte Datei wiederhergestellt.

Kommunikation

Die Übertragung der Shares vom Clientrechner zu den Storageservern und umgekehrt erfolgt unter Einsatz von Protokolle wie TLS oder IPsec. Der eingesetzte Verschlüsselungsalgorithmus ist variabel; dabei sind folgende Konfigurationen möglich:

fixed Bei jeder Sitzung wird derselbe, vorher festgelegte Algorithmus verwendet.

random Bei jeder Sitzung wird zufällig und gleichverteilt ein Algorithmus aus einer festgelegten Menge möglicher Algorithmen gewählt.

distinct Bei jeder Sitzung wird garantiert ein anderer Algorithmus verwendet. So kann beispielsweise auch beim Hochladen der Shares auf die Storageserver für die Verbindung zu jedem einzelnen Storageserver ein unterschiedlicher Algorithmus zum Verschlüsseln eingesetzt werden.

Der Nutzer des eSafe kann das System entsprechend konfigurieren und damit an sein persönliches Sicherheitsbedürfnis anpassen.

Langfristige Sicherheit

Durch das beschriebene Speicherkonzept wird erreicht, dass die Vertraulichkeit der gespeicherten Daten nicht von der Sicherheit eines Kryptosystems abhängt. Diese Sicherheit ist in der Regel zeitlich beschränkt und abhängig von der Wahl der verwendeten Schlüssellängen. Das beschriebene System ist damit in besonderem Maße für eine langfristige Speicherung elektronischer Daten geeignet. Aus diesem Grund kann der eSafe als Langzeitspeicher in die vom Bundesamt für Sicherheit in der Informationstechnik entwickelte Referenzarchitektur zur vertrauenswürdigen Langzeitarchivierung [Bun09] integriert werden.

Für die langfristige Lesbarkeit der gespeicherten Dokumente spielt die Wahl geeigneter Datenformate eine zentrale Rolle. Diese sollten standardisiert und langlebig sein. Als für die Langzeitaufbewahrung geeignete Formate gelten unter anderem PDF/A sowie XML. Die Wahl geeingneter Datenformate liegt in der Verantwortung des eSafe-Nutzers.

Weiterhin ist zu beachten, dass elektronische Signaturen ebenfalls einem Alterungsprozess unterliegen und mit der Zeit ihre Beweiskraft verlieren können (vgl. [BPRS02, Arcb, Arca]). Zum Erhalt qualifizierter Signaturen fordert §17 der Signaturverordnung vor Ablauf der Sicherheitseignung der verwendeten Algorithmen und Parameter eine neue qualifizierte Signatur, die die bisherigen Signaturen einschließt, sowie einen qualifizierten Zeitstempel [Sig]. Sofern der verwendete Hash-Algorithmus noch als sicher gilt, reicht es aus, lediglich die bisherigen Signaturen mit einem qualifizierten Zeitstempel zu versehen [BPRS02]. Stellt der Nutzer ein signiertes Dokument in seinen eSafe ein, so wird er durch

das System informiert, sobald die Signatur erneuert werden muss. Wünschenswert wäre es, dass für die Erneuerung der Signatur nicht die betroffene Datei rekonstruiert werden muss, sondern lediglich die entsprechenden Shares neu signiert werden. Lösungen für dieses Problem werden derzeit innerhalb des Forschungsprojekts entwickelt.

Literaturverzeichnis

[Arca] ArchiSafe. http://www.archisafe.de/, last checked 24.04.2009.

[Arcb] ArchiSig – Beweiskräftige und sichere Langzeitarchivierung digital signierter Dokumente. http://www.archisig.de/, last checked 24.04.2009.

[BdI08] Bundesministerium des Innern. Einführung des elektronischen Personalausweises in Deutschland. Grobkonzept – Version 2.0, Juli 2008. `http://213.216.17.150/DOL/Anlagen/Anlage_A13_ePA_Grobkonzept2.0.pdf`.

[BPRS02] Ralf Brandner, Ulrich Pordesch, Alexander Roßnagel, and Joachim Schachermayer. Langzeitsicherung qualifizierter elektronischer Signaturen. *DuD*, 26, 2002.

[Bun09] Bundesamt für Sicherheit in der Informationstechnik. Vertrauenswürdige elektronische Langzeitarchivierung (VLA). Technische Richtlinie VLA (BSI-TR-03125), in Vorbereitung, 2009.

[MDNK08] Toshiyuki Miyamoto, Shinji Doi, Hiroki Nogawa, and Sadatoshi Kumagai. Autonomous distributed secret sharing storage system. *Systems and Computers in Japan*, 37(6):55–63, 2008.

[Sha79] Adi Shamir. How to share a secret. *Commun. ACM*, 22(11):612–613, 1979.

[Sig] Verordnung zur elektronischen Signatur, sigv. `http://bundesrecht.juris.de/sigv_2001/index.html`, last checked 24.04.2009.