ReEncryption — Das Konzept für den umfassenden Dokumentenschutz

Stephen D. Wolthusen[1] · Frank Prediger[2]

[1]Fraunhofer-IGD, Darmstadt
wolt@igd.fhg.de

[2]ReEncryption GmbH, Darmstadt
frank.prediger@reencryption.de

Zusammenfassung

Dieser Beitrag beschreibt ein IT-Sicherheitssystem, das die Bearbeitung, Speicherung und Übertragung von vertraulichen oder anderweitig schützenswerten Datensätzen und Dokumenten innerhalb einer geschlossenen Nutzergruppe auf Grundlage bestehender kommerzieller Betriebssysteme ermöglicht.
Kryptographische Schutzmechanismen gewährleisten dabei Vertraulichkeit und Integrität von Dokumenten in digitaler Repräsentation; auch analoge Repräsentationen wie Ausdrucke sind durch den Einsatz digitaler Wasserzeichen geschützt.
Die Verwaltung, insbesondere aber die Kontrolle über die Nutzung von Dokumenten ist in das System derart integriert, daß jede Verwendung durch einen Nutzer kontrolliert und protokolliert werden kann; aufgrund automatischer und transparente Verschlüsselung kann selbst bei Speicherung auf nicht-überschreibbaren Datenträgern durch Sperrung des zum Zugriff erforderlichen Schlüsselmaterials die weitere Verwendung unterbunden werden.
Die hierbei verwendeten Mechanismen sind für Anwendungsprogramme und Nutzer transparent und können durch diese nicht umgangen werden, da die Schutzmaßnahmen innerhalb des Betriebssystems (sowohl Microsoft Windows als auch Unix-Derivate) verankert sind.

1 Einleitung

Obwohl in IT-Systemen vorgehaltene Daten sowie die Kommunikation vernetzter Systeme kritische Ressourcen darstellen und Unternehmen häufig zum Schutz dieser Ressourcen die Ergebnisse Sicherheitsrichtlinien definieren, erfolgt die Einhaltung der in Sicherheitsrichtlinien festgelegten Richtlinien nicht immer in wünschenswertem Umfang.

Gründe für die Nichtbefolgung sind wohlbekannt, so werden z.B. eventuell erforderliche kryptographische Sicherungsmaßnahmen für die Sicherstellung der Vertraulichkeit elektronischer Kommunikation nur punktuell durchgeführt, da diese zusätzliche Arbeitsschritte darstellen die aus Nachlässigkeit oder aufgrund anderer Prioritäten wie Zeitdruck nicht erfolgen.

Dies gilt selbst dann, wenn die Mechanismen grundsätzlich von der IT-Infrastruktur bereitgestellt werden, deren Einsatz jedoch dem Ermessen einzelner Mitarbeiter anheim gestellt ist, da die Einhaltung der Sicherheits-Richtlinien zumindest im kommerziellen Umfeld meist eine nachgeordnete Zielstellung ist.

Wider besseren Wissens werden daher insbesondere vertrauliche Informationen ohne Vertraulichkeitsmechanismen per Email über öffentliche Netze versandt, oder auch Datenträger und Rechner, die vertrauliche Daten enthalten, ungeschützt potentiellen Angreifern leicht zugänglich gemacht.

Insbesondere bei Dokumenten (z.B. Kalkulationen, F&E-Ergebinsse, Patientendaten, Vorstandsbeschlüsse, aber auch Rechnungen oder Frachtbriefe) bestehen neben Anforderungen an Vertraulichkeit weitere Bedürfnisse nach Charakteristiken wie etwa der überprüf- und nachweisbaren Integrität von Dokumenten (z.B. bei Frachtbriefen), der Authentizität, aber auch der Revisionssicherheit elektronisch abgelegter Dokumente.

1.1 Existierende Ansätze

Für die zuvor genannten Anforderungen insbesondere im Bereich der Sicherstellung von Vertraulichkeit, Integrität und Authentizität existieren sowohl im Forschungsbereich als auch in Produktform für die meisten relevanten Plattformen eine Reihe von Lösungen, die sowohl auf Betriebssystemebene als auch in Anwendungsprogrammen diese Mechanismen sicherstellen.

Für die Vertraulichkeit von Daten auf nichtflüchtigen Datenträgern existieren z.B. blockbasierte Verschlüsselungsmechanismen wie Utimaco Safeguard Easy, Ansätze, die Dateisysteme in Dateien simulieren wie etwa PGPdisk der PGP Corporation oder Microsoft EFS, sowie Schichtungsmechanismen für Dateisysteme wie CFS (Blaze 1993), TCFS (Mauriello 1997), oder die FiST-Familie von Mechanismen (Zadok 2001).

Diese Mechanismen bieten jedoch einerseits keinen Integritätsschutz der gesicherten Daten, sodaß ein Angreifer, der den Chiffretext eines verschlüsselten Datenträgers modifizieren kann, den Klartext ohne Kenntniss des Schlüssels (in einer ihm zunächst unbekannten Weise) modifizieren kann. Andererseits ist der Zugriffsschutz auf derartige Daten auf die Kenntnis oder Verfügbarkeit des Schlüsselmaterials selbst begrenzt, da darüber hinaus keine Zugriffskontrollmechanismen in den genannten Systemkategorien vorhanden sind.

Dokumenten-Verwaltungssysteme bieten zwar zumeist eigene Sicherheitsmechanismen, doch sind diese auf die zentralen Datenhaltungs-Mechanismen beschränkt (Bertino und Ferrari 2002); sofern auch auf Endgeräten eine Kontrolle über die Nutzung von Dokumenten erfolgt, so sind diese häufig nicht nur gezielt sondern insbesondere auch unbeabsichtigt zu unterlaufen, sodaß nach der Ausgabe eines Dokuments an ein Endgerät bzw. den Nutzer dieses Rechners kaum weitere Kontrolle über die Weiterverarbeitung des Dokumentes oder davon abgeleiteter Datensätze bestehen.

Im Bereich der Netzwerkverkehrs-Absicherung existieren bereits eine Reihe von Standards, die direkt in Produkten – insbesondere kommerziell und frei verfügbaren Betriebssystemen – enthalten sind, so etwa IPSec (Thayer, Doraswamy und Glenn 1998) für die Bereitstellung von Virtual Private Networks (VPN).

Obwohl IPSec grundsätzlich für die Bereitstellung von Ende-zu-Ende Sicherheitsmerkmalen ausgelegt ist, beschränkt sich jedoch der Einsatz meist auf VPN-Gateways, da im Bereich der Infrastrukturen und Schlüsselaustauschmechanismen (Harkins und Carrel 1998, Busch und Wolthusen 2002) zum Teil noch erhebliche Interoperabilitätsprobleme bestehen. Dadurch sind jedoch wesentliche Teile des Netzwerkverkehrs nach wie vor auch bei Einsatz von VPN-Mechanismen an Übergangsknoten verwundbar.

Darüber hinaus existieren eine Reihe von Protokollen wie etwa das Transport Layer Security Protocol (Dierks und Allen 1999), die auf der Transport-Ebene des OSI-Referenzmodells Vertraulichkeit, Integrität, und Schutz vor Wiedereinspielung bereitstellen.

Die Bereitstellung dieser Sicherheitsmerkmale für den Bereich Email erfolgt ebenfalls günstigstenfalls (d.h. bei Integration in den jeweiligen Mail User Agent (MUA)) auf Anwendungsebene und kann so eine Ende-zu-Ende Sicherheit bereitstellen; die Mehrzahl der verfügbaren MUA besitzen derartige Funktionalität.

Auch hier ist jedoch wiederum festzuhalten, daß die Verwendung der entsprechenden Merkmale im Ermessen des jeweiligen Anwenders liegt, und aufgrund der vorzunehmenden zusätzlichen Operationen (so etwa dem manuellen Austausch oder die Verifikation von Schlüsselmaterial für kryptographische Verfahren) nicht immer durchgeführt werden (Whitten und Tygar 1998).

2 Anforderungen

Aufgrund der vorangegangenen Diskussion lassen sich für effektive und effiziente Mechanismen für die Gewährleistung von Sicherheits-Eigenschaften eine Reihe von Anforderungen ableiten, die im Folgenden kurz dargestellt werden.

- Sicherheitsmechanismen müssen für Nutzer weitestgehend unsichtbar und automatisch realisiert sein. Durch die zwingende, automatische Umsetzung entsprechender Vorgaben bezüglich der Sicherheit von Daten und Nachrichten lassen sich einerseits sicherheitsrelevante auf Fahrlässigkeit beruhende Unterlassungen und Fehlbedienungen eliminieren, andererseits ergeben sich aus einer derartigen Realisierung Effizienzgewinne dadurch, daß Mitarbeiter keine nicht unmittelbar aufgabenbezogene Handlungen zur Gewährleistung der Sicherheit durchführen müssen und zudem kein Schulungsaufwand für die Bedienung der Sicherheitsmechanismen anfällt.
- Sicherheitsmechanismen müssen ebenfalls für Anwendungsprogramme weitestgehend unsichtbar sein, insbesondere wenn besagte Programme selbst keine hinreichenden Sicherheitsmechanismen verfügen. Dadurch lassen sich Anwendungsprogramme von Drittherstellern bei erhöhter Sicherheit weiterverwenden und die Aufwände für die Anpassung von Altprogrammen aus Eigen- oder Auftragsentwicklung für die anwendungsspezifische Integration von Sicherheitsmechanismen minimieren.
- Insbesondere im kommerziellen Umfeld müssen Sicherheitsmechanismen am Dokumentenmodell orientiert sein und die Definition von Sicherheitseigenschaften sowie deren Durchsetzung von der Erstellung eines Dokumentes bis zu dessen Vernichtung oder Archivierung unterstützen. Dies muß ungeachtet des jeweiligen Speicherungsortes erfolgen und zudem jederzeit konsistent und ungeachtet der Zugriffmechanismen der Fall sein.
- Der Zugriff auf geschützte Dokumente sowie insbesondere auf von diesen abgeleiteten Objekten muß jederzeit durch die Sicherheitsmechanismen kontrolliert werden; dies gilt insbesondere auch, sofern sich diese Daten aufgrund der Speicherung auf nichtflüchtigen Datenträgern oder der Übertragung über Kommunikationsschnittstellen außerhalb der unmittelbaren Kontrolle der Sicherheitsmechanismen befinden.
- Sicherheitsmechanismen für Dokumente müssen ein feinkörniges Berechtigungsmodell bereitstellen, mit denen für Geschäftsprozesse relevante Zugriffsrechte effektiv dargestellt

werden können. Hierzu muß es möglich sein, Operationen auf Dokumenten oder von Dokumenten abgeleiteten Objekten jederzeit einem bestimmten Nutzer zuzuordnen und so das Berechtigungsmodell anwenden zu können.

- Die exakte Verwendung von Dokumenten durch berechtigte Nutzer, ungeachtet der verwendeten Repräsentation sowie der hierzu verwendeten Anwendungsprogramme muß jederzeit nachvollziehbar protokolliert werden können.
- Es müssen insbesondere auch Sicherheitsmechanismen bereitgestellt werden, welche bei der Transformation von Dokumenten in analoge Repräsentation (z.B. Ausdrucke) durch berechtigte Personen weiterhin die Verfolgung der Quelle und Identität des Dokumentes ermöglichen, um so bei Verletzung von Sicherheitsrichtlinien den Ursprung eines fehlgeleiteten Dokumentes identifizieren zu können.

Darüber hinaus bestehen noch eine Reihe weiterer Anforderungen, die sich aus den jeweiligen Einsatzgebieten (z.B. Revisionssicherheit von archivierten Datensätzen) sowie aus rechtlichen Anforderungen (Art und insbesondere Zugänglichkeit der Protokoll- und Revisionsdaten sind so z.B. aufgrund von Anforderungen des BetrVerfG sowie des BDSG einzuschränken oder gesondert zu schützen).

Eine Sicherheitsarchitektur, welche die hier genannten Anforderungen erfüllt, wird in Abschnitt 3 vorgestellt.

3 Architektur

Das ReEncryption-System stellt eine integrierte Sicherheitsarchitektur dar, die insbesondere zur Verwaltung, Verfolgung, und Zugriffskontrolle vertraulicher oder anderweitig (z.B. urheberrechtlich) schützenswerter Daten und Dokumente dar.

3.1 Randbedingungen

Der Entwurf der Architektur geht explizit davon aus, daß es sich um ein System für den Einsatz in kontrollierten Umgebungen (z.B. im kommerziellen Sektor, etwa im Bereich der Forschung & Entwicklung oder des Personalwesens) handelt, bei dem die physische Sicherheit der Rechner sichergestellt ist und Nutzer, deren Verhalten durch das ReEncryption-System überwacht und kontrolliert werden, keine administrativen Privilegien an den jeweiligen Systemen besitzen oder erlangen können.

In dem vorgesehenen Anwendungsszenario ist (in Ergänzung zu den in Abschnitt 2 genannten Anforderungen) gefordert, daß geistiges Eigentum im Regelfall innerhalb einer geschlossenen Nutzergruppe (z.B. eines Unternehmens) geschützt innerhalb eines wohldefinierten unternehmensweiten IT-Struktur verarbeitet, gespeichert, und übertragen werden müssen; die IT-Struktur kann sich dabei über eine große Anzahl vernetzter Rechner an verschiedenen Standorten erstrecken.

Innerhalb der so definierten geschlossenen Nutzergruppe werden nutzerdefinierte personenbezogene und gruppen- bzw. rollenbezogene differenzierte Rechte als angemessen angesehen; zwingend durchgesetzte Sicherheitsmechanismen betreffen dabei jedoch den potentiellen Austausch mit Personen oder Rechnern außerhalb der geschlossenen Nutzergruppe; letzterer ist

nur designierten Personen mit besonderen Rechten für die Freigabe von Dokumenten für den Austausch vorbehalten.

Unter Zugrundelegung dieser Annahmen werden Sicherheitsmechanismen in den Betriebssystemkern sowie weitere vor nicht privilegierten Nutzern geschützten Komponenten derart eingefügt, daß kritische Pfade in und aus einem geschützten Rechner unter der Kontrolle des Sicherheitssystems stehen und zudem das Sicherheitssystem selbst vor möglichen Manipulationen und Umgehungsversuchen seitens der Nutzer oder auch Anwendungsprogrammen geschützt ist.

Die Durchsetzung der Sicherheitsanforderungen für Speichermedien (gleich, ob lokale, Wechseldatenträger, oder Netzwerk-Speichermechanismen) sowie für Netzwerk-Datenverkehr kann durch die Verwendung von Verschlüsselungsmechanismen realisiert werden, die derart in den jeweiligen Datenstrom eingefügt werden, daß diese für legitime Nutzer und Anwendungsprogramme vollständig transparent bei Verlassen des Rechners zwingend verschlüsselt und auf legitimen Systemen mit Zugriff auf das geeignete Schlüsselmaterial ebenso wieder transparent entschlüsselt werden.

Dadurch wird implizit die geschlossene Nutzergruppe aufrecht erhalten, insbesondere auch dann wenn die Sicherheitsmechanismen selbst auf dem gesschützten Rechner außer Betrieb sind. Dies kann z.B. dann der Fall sein, wenn Speichermedien eines Rechners mit einem fremden Betriebssystem bearbeitet werden, welche a priori die Sicherheitsmerkmale selbst nicht beachtet.

3.2 Zentrale Rechteverwaltung und Schlüsselvergabe

Um derartige Angriffe deutlich zu erschweren, selbst wenn ein geschützter Rechner in den Besitz eines Dritten gelangt (z.B. bei Verlust eines Notebook-Rechners), wird dabei das erforderliche Schlüsselmaterial nicht persistent auf dem geschützten Rechner gespeichert, sondern vielmehr auf Anforderung — dies geschieht implizit bei Zugriff auf geschützte Datensätze, gleich ob aus Dateisystemen oder über Netzwerk-Datenverkehr — von einem vertrauenswürdigen System über einen ebensolchen Kanal an den geschützten Rechner weitergeleitet. Die Übermittlung des Schlüsselmaterials erfolgt nur dann, wenn der Nutzer zum Zeitpunkt des Zugriffs das Recht besitzt, auf diesen Datensatz zuzugreifen.

Sofern der Verdacht entsteht, daß ein Rechner z.B. entwendet und potentiell noch die I&A-Daten eines Nutzers kompromittiert wurden, kann dabei durch Eintragung des als kompromittiert vermuteten Rechners in eine Certificate Revocation List implizit die Auslieferung des erforderlichen Schlüsselmaterials vollständig verhindert werden.

Das Anwendungsszenario geht davon aus, daß an einem Rechner zu einem gegebenen Zeitpunkt nur ein einzelner legitimer Nutzer aktiv sein kann[1]; zu diesem Zweck muß sich der Nutzer (zunächst orthogonal zu den bestehenden I&A-Mechanismen des Betriebssystems, jedoch ist die Integration in Single-Sign-On-Mechanismen möglich) gegenüber dem Sicherheitssystem identifizieren und authentisieren. Dieser Vorgang erfolgt gegenüber einer zentralen Instanz — dem Key Center — welche die Nutzeridentitäten und Privilegien verwaltet, Abweichungen von Nutzungsmodellen (z.B. die Präsenz eines Nutzers an mehreren Rechnern gleichzeitig, der nur

[1] Die Einschränkung auf einen einzelnen aktiven Nutzer zu einem gegebenen Zeitpunkt ergibt sich aus der Verfügbarkeit und hohen Kapazität von verdeckten Kommunikationskanälen in den zugrundeliegenden Betriebssystemen.

durch Wissen authentisiert ist) erkennen und geeignet reagieren und zudem Revisionsdaten und Profile erstellen kann. Das Key Center ist ebenfalls dafür verantwortlich, die dokumentenspezifischen Rechte durchzusetzen. Da die Kommunikationskomplexität der Rechte-Überprüfung für jeden einzelnen Zugriff auf Ressourcen in Dateisystemen und insbesondere im Netzwerk-Datenverkehr gegen eine zentrale Rechteverwaltung deutlich zu hoch wäre, ist es erforderlich hierbei eine geeignete Unterscheidung zu treffen, um diese Komplexität zu reduzieren.

Zu diesem Zweck werden Verschlüsselungsmechanismen dazu verwendet, neu erzeugte (oder auch aus bestehenden abgeleitete) Datensätze an einen bestimmten Rechner bzw. an einen bestimmten Nutzer zu koppeln[2]. Sofern jedoch Datensätze mit anderen Nutzern ausgetauscht werden sollen, ist es erforderlich, diese dem zentralen Zugriffs- und Verwendungskontrollsystem bekannt zu geben. Dieser Registrierungsprozeß verfolgt mehrere Ziele. Ein registriertes Dokument wird eindeutig[3] (anhand einer kryptographischen Hash-Funktion) identifiziert und mit einer Reihe von Meta-Informationen assoziiert. Diese beinhalten den Namen eines Dokumentes sowie einen Zugriffspfad (Uniform Resource Identifier, URI) innerhalb eines virtuellen hierarchischen Dateisystems als sekundäres (anders als das primäre Merkmal veränderliche) Identifikationsmerkmal, sowie weitere Informationen wie Schlüsselworte und Klassifizierungen.

Insbesondere sind jedoch in den Meta-Informationen Zugriffs- und Nutzungsinformationen enthalten, welche zunächst zum Zeitpunkt der Registrierung eines jeden Dokumentes gesetzt, anschließend jedoch von berechtigten Nutzern geändert werden können. Diese Rechte geben den Besitzer eines Dokumentes an sowie eine Menge von Gruppen- und allgemeinen Rechten. Die gruppenbezogenen Rechte sind in Form eines gerichteten azyklischen Graphen organisiert und können so beliebige strukturierte Mengen oder auch einzelne Nutzer mit Rechten wie sie in Arbeitsabläufen häufig benötigt werden, effektiv darstellen. Darüber hinaus werden je Dokument auch Informationen vorgehalten, welche bei Bedarf Zeitfenster für die Verfügbarkeit auch berechtigter Nutzer angeben.

Die so registrierten Dokumente werden optional auf sogenannten Content Servern in Datenbanken vorgehalten, die einen effektiven Zugriff auf Dokumente und Meta-Informationen ermöglichen, jedoch nicht zwingend erforderlich sind. Der Prozeß der Registrierung eines Dokumentes beinhaltet darüber hinaus die Anbringung einer digitalen Signatur des Nutzers, der die Registrierung vornimmt sowie einen Zeitstempel, der gleichzeitig ein Revisionsereignis darstellt. Damit können Dokumente, die einmal registriert wurden, nicht mehr verändert oder verfälscht werden. Um dennoch eine Wiederverwendung derselben URI zu ermöglichen, sind für eine URI mehrere Versionen zulässig[4]

3.3 Schutzmechanismen für registrierte Dokumente

Um die Integrität und Vertraulichkeit der registrierten Dokumente sicherstellen zu können, werden diese bei geeignetem Datentyp mit einem digitalen Wasserzeichen (siehe unten) versehen und mit Schlüsseln, die vom Key Center jeweils für ein Tupel $\langle \text{Nutzer}_i, \text{Dokument}_j \rangle$ spezifisch vergeben werden, verschlüsselt und zusammen mit integritätsgeschützten Metadaten abgelegt.

[2] Dies stellt einen weiteren Grund für die Einschränkung auf einen einzelnen aktiven Nutzer je Rechner dar, da so erhebliche Geschwindigkeitsvorteile erzielt werden können.

[3] Unter Vernachlässigung der Wahrscheinlichkeit einer Kollision.

[4] Diese werden analog zum Namensschema in OpenVMS mittels Semikolon und Versionsnummer identifiziert; eine URI ohne Versionsnummer ist implizit die höchste verfügbare Versionsnummer (McCoy 1990).

Sofern nun von einem Content Server oder einem beliebigen anderen Ort über Netzwerk- oder Dateisystems-Operationen Zugriffe auf registrierte Dokumente erfolgen, so erfolgt dies ausschließlich in verschlüsselter Form und zusammen mit den zur Erlangung der Zugriffsberechtigung (d.h. des Schlüsselmaterials) erforderlichen Meta-Informationen. Die Entschlüsselung sowie die Einbettung weiterer digitaler Wasserzeichen bei geeignetem Dokumententyp erfolgt dabei unabhängig von dem Pfad, den ein derartiges registriertes Dokument genommen hat durch Betriebssystem-Erweiterungen, welche ausschließlich für die Bereithaltung des Dokuments im (flüchtigen) Arbeitsspeicher sorgen.

Dadurch ist der Sicherheitsmechanismus einerseits von spezifischen Anwendungsprogrammen und Übertragungswegen unabhängig und andererseits auch nicht durch diese oder sonstige Operationen seitens des Nutzers zu unterbinden; konzeptuell ähnliche Ansätze (Gifford, Jouvelot, Sheldon und O'Toole, Jr. 1991, Dourish, Edwards, LaMarca, Lamping, Petersen, Salisbury, Terry und Thornton 2000) für die verteilte Integration von Metadaten sind dahingehend zumeist an Anwendungsprogramme gebunden.

Um sicherzustellen, daß Informationskanäle möglichst schmale Bandbreite aufweisen werden dabei Dateien, die von einem Nutzer mit Schreibrechten geöffnet werden, automatisch mit einem für diesen Nutzer spezifischen Schlüssel verschlüsselt und somit ungeachtet des Speicherungsortes (z.B. ein nicht mit den hier beschriebenen Betriebssystem-Erweiterungen ausgestatteter Server) gesichert. Sofern jedoch für eine von einem Nutzer abgespeicherte Datei der kryptographische Hashwert mit dem eines registrierten Dokumentes übereinstimmt, so wird dieses in die oben beschriebene Form transformiert und somit für andere berechtigte Nutzer — wiederum ungeachtet des Speicherungsortes oder Übertragungsmechanismus' — zugänglich. Dadurch wird insbesondere auch implizit erreicht, daß für jeden Zugriff eines Nutzers eine Anforderung von Schlüsselmaterial erfolgt, was wiederum eine Prüfung der Nutzungsrechte und der Vertrauenswürdigkeit des Nutzers sowie des Systems, von dem aus die Anforderung erfolgt, bedingt. Dadurch können jedoch jederzeit verzögerungsfrei zentral Rechte vergeben werden und ungeachtet, ob evtl. ein Nutzer eine Datei auf einem schreibgeschützten Datenträger ein registriertes Dokument abgespeichert hat Rechte widerrufen und durchgesetzt (z.B. durch das o.g. Nutzungs-Zeitfenster) werden.

Sofern daher ein Dokument von einem Nutzer bearbeitet oder auch aus bestehenden registrierten Dokumenten abgeleitet wurde, ist es zum Austausch dieser Dokumente mit anderen Nutzern erforderlich, dieses zu registieren und somit unter anderem diesbezügliche Revisionsdaten zu erzeugen.

3.3.1 Digitale Wasserzeichen

Da es eine meist nicht hinnehmbare Einschränkung der Arbeitsabläufe bedeuten würde, Ausdrucke zu verbieten (oder aber auch sonstige Transformationen in analoge Daten wie etwa das abphotographieren von Bildschirminhalten oder aber die Aufnahme von Tondaten durch analoge Ausgänge eines Rechners), sind Schutzmechanismen auch im analogen Bereich wünschenswert. Dies kann mit Hilfe der zuvor genannten digitalen Wasserzeichen (Cox, Miller und Bloom 2002, Arnold, Schmucker und Wolthusen 2003) realisiert werden.

Dabei werden einerseits wie zuvor erwähnt zum Zeitpunkt der Registrierung eines Dokumentes mehrere digitale Wasserzeichen parallel eingebracht, welche den Ursprung des Dokumentes sowie dessen Identität in Form eines partiellen Hashwertes enthalten. Andererseits wird für registrierte Dokumente vor der Weitergabe aus dem Betriebssystemkern an Anwendungsprozesse

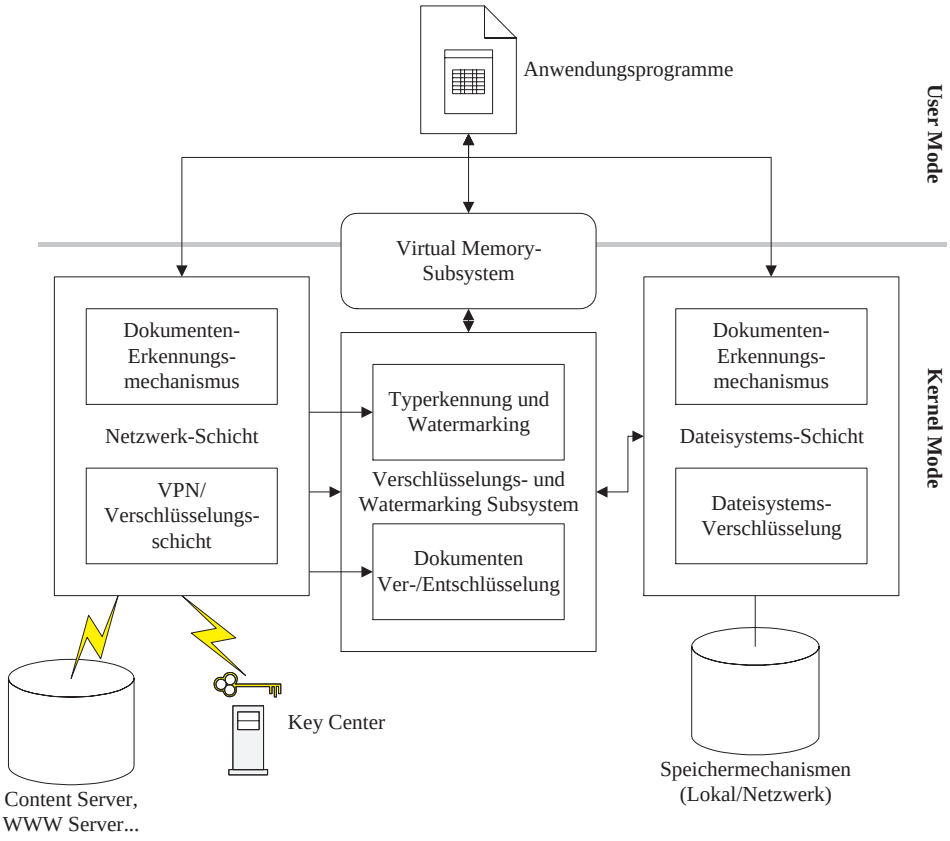

Abb. 1: Wasserzeichen-Einbettung in geschlützten Systemen

ein weiteres Wasserzeichen eingebracht, welches die Identität des aktiven legitimen Nutzers beinhaltet (siehe Abbildung 1).

Zwar ermöglichen digitale Wasserzeichen nicht die Prävention einer Fehlhandhabung (gleich ob durch bewußte Handlungen eines Nutzers oder etwa durch Fahrlässigkeit verursacht), sondern können nur einerseits einen Abschreckungseffekt erzielen und andererseits die gezielte Rückverfolgung von an ungewünschter Stelle vorgefundenen Dokumenten erlauben.

Darüber hinaus ist jedoch die Einbettung der Dokumenten-Identifikation aufgrund der Robustheits-Eigenschaften der digitalen Wasserzeichen von Interesse; so ist es z.B. selbst bei Vorliegen eines Bild-Fragmentes von nur 10% der ursprünglichen Fläche möglich, die Identität des Originals zu bestimmen und somit das vollständige digitale Original von einem geeigneten Speicherungsort wiederzufinden.

4 Aspekte der Implementierung

Das hier beschriebene System gliedert sich in drei Klassen von Komponenten, geschützte Rechner, an denen Nutzer direkt arbeiten, sowie das Key Center sowie zum dritten die Content Server, mit denen Nutzer nur mittelbar interagieren.

4.1 Geschützte Rechner

Um eine möglichst breite Basis von Einsatzgebieten abdecken und dabei bestehende Anwendungsprogramme unverändert weiterverwenden zu können, wurden die Sicherheitsmechanismen nachträglich, und ohne dabei Zugriff auf oder Modifikationen an Quellcode des Betriebssystems zu erfordern, in die Microsoft Windows NT-Familie (2000,XP) von Betriebssystem eingebettet um so geschützte Rechner zu erhalten.

Die einzelnen möglichen Kommunikationskanäle eines geschützten Rechners wurden durch jeweils komponentenspezifische Erweiterungen abgesichert. So wurde mit Hilfe eines Dateisystems-Filtertreibers der Zugriff auf sämtliche Dateisysteme ungeachtet ob lokaler Art oder im Netzwerk realisiert und zudem durch die Positionierung auf einer Abstraktionsebene oberhalb des Dateisystems die Dateisemantik erhalten.

Erforderliche kryptographische Operationen sowie die Einbettung von digitalen Wasserzeichen erfolgen dabei mit Hilfe eines zentralen Dienstes, welcher auch die Abstraktion von den jeweils verwendeten Implementierungen und Algorithmen (siehe Abschnitt 5) gestattet. Zudem bietet dieser zentrale Dienst die Möglichkeit, bekannte (im Arbeitsspeicher befindliche) registrierte Dokumente zu erfassen und bei Speicherung zu erkennen.

Analog hierzu existieren auch für weitere Geräteschnittstellen entsprechende Filterungsmechanismen, welche jeweils die Erkennung der Datenströme sowie möglicherweise erforderliche Anforderungen von Schlüsselmaterial und kryptographische Operationen an den zentralen Dienst delegieren.

Die Netzwerk-Schnittstellen des Systems sind in zweierlei Hinsicht für das hier beschriebene System relevant. Einerseits wird auf der Transportebene im Protokollstapel für Anwendungsprotokolle unsichtbar eine Sicherung auf Grundlage des TLS-Protokolls realisiert, welche Vertraulichkeit, Integritätsschutz, sowie Schutz vor Wiedereinspielung implizit und für die Nutzer der Netzwerkschnittstelle transparent realisiert. Der Verbindungsaufbau erfolgt dabei zwingend unter Verwendung starker wechselseitiger Authentisierung, womit die Identifikation und Authentisierung des geschützten Rechners gegenüber anderen Systemen, insbesondere jedoch aber gegen das Key Center (und somit auch die Erzwingung des Ausschlusses kompromittierter Systeme durch Einsatz von Certificate Revocation Lists) realisiert wird. Diese Sicherungsschicht ist dahingehend konfigurierbar, daß für jeden beliebigen Host (oder aber ein Subnetz) von Seiten der Rechnerverwaltung angegeben werden kann, ob Kommunikation a priori erlaubt ist, diese im Klartext erfolgen kann, oder ob die zuvor genannten transparenten kryptographischen Schutzmechanismen eingesetzt werdem müssen.

Andererseits existiert analog zu dem Mechanismus für Dateisysteme ein weiterer Mechanismus, der Datenströme auf das mögliche Enthaltensein von Dokumenten untersucht und bei Erkennung transparent durch entschlüsselte und mit digitalen Wasserzeichen versehene Dokumente ersetzt[5]. Auch dieser Mechanismus ist auf Transportebene im OSI-Referenzmodell angesiedelt, da dort bereits ein in korrekte Abfolge gebrachter Datenstrom vorliegt und nicht die Notwendigkeit existiert, bei paketbasierten Netzen eine Defragmentierung, Fehlerkorrektur, und Sortierung vornehmen zu müssen.

[5] Dieser Mechanismus ist durch die Modifikation des Datenstroms trivial zu umgehen, was jedoch für eingehende Datenströme keine Bedrohung darstellt, da bei Nichterkennung der Datenstrom nicht entschlüsselt wird, somit ein Angreifer keinen Vorteil erlangt.

Dennoch muß mit Hilfe eines *sliding window* der Datenstrom auf den Beginn eines registrierten Dokumentes untersucht werden und dieser transparent durch das Klartext-Äquivalent ersetzt werden. Dies ist insbesondere daher interessant, da durch die Entfernung der Metadaten und Entfernung des eventuell vorhandenen Paddings der verwendeten Chiffre sich eine Diskrepanz zwischen den Längen des Originals und Substitus ergeben (Rademer und Wolthusen 2001).

Die Einbettung von digitalen Wasserzeichen ist abhängig von dem jeweils vorliegenden Medientyp (z.B. Bild- oder Tondaten, strukturierter Text, oder hybride Konstrukte) und erfordert jeweils typspezifische Wasserzeichen-Verfahren. Um hierbei flexibel und erweiterbar zu sein, beinhaltet das Watermarking-Subsystem eine Reihe von Typdetektoren, welche einen von einer der oben genannten Komponenten erhaltenen Datenstrom parallel auf den eigenen Medientyp untersuchen und bei Erfolg geeignet Wasserzeichen einbetten können (Busch und Wolthusen 2001).

Die Identifikation und Authentisierung erfolgt bei der Realisierung unter Microsoft Windows NT/2000/XP in Form eines Moduls, das in das Graphical Identification and Authentication Subsystem (GINA) eingefügt wird; damit kann sowohl eine Single-Sign-On als auch eine vollständig orthogonale I&A realisiert werden.

4.2 Key Center

Key Center sind aufgrund ihrer zentralen Rolle, der erforderlichen Ausfallsichereheit und Skalierbarkeit in Form einer mehrschichtigen Architektur aufgebaut.

Die eigentliche Datenhaltung erfolgt in Form einer replizierten und parallel betriebenen Datenbank, welche sowohl Ausfallsicherheit bei Versagen eines einzelnen Datenbank-Systems als auch Lastausgleich zwischen den Knoten bei gleichzeitiger Synchronität aller so erfolgenden Transaktionen gewährleisten. Die Kommunikation mit den in Abschnitt 4.1 beschriebenen Rechnern erfolgt dabei jeweils über einen Cluster von Rechnern, welche die erforderliche Logik für die Operationen des Key Center beinhalten und transaktionsorientiert operieren.

Zur Sicherstellung des geschützten Kanals zu den Endgeräten, aber auch zu Content Servern (siehe Abschnitt 4.3) wird der bereits in Abschnitt 4.1 beschriebene Mechanismus verwendet. Insbesondere kann an dieser Stelle jedoch die Synchronisation und Propagierung mit Certificate Revocation Lists erfolgen und implizit im Rahmen anderweitiger Kommunikation an weitere Rechner übergeben werden.

4.3 Content Server

Content Server stellen Speicherungsmechanismen für registrierte Dokumente zur Verfügung und können gleichzeitig als Schnittstelle für die Verwaltung von Metadaten dieser Dokumente verwendet werden; der interne Aufbau ist dabei analog dem im vorigen Abschnitt beschriebenen (Dokumente können dabei bei Bedarf in Datenbanken oder auch in Dateisystemen abgelegt sein).

Zur Gewährleistung der Integrität des Dokumentenbestandes (z.B. Verhindeung der gleichzeitigen Registierung desselben Dokuments durch unterschiedliche Nutzer oder auch Rechtekonflikte) wickeln dabei Content Server verschachtelte Transaktionen mit dem Key Center ab, um so dem Nutzer eine konsistente Sicht zu gewährleisten.

5 Ausblick

Das in diesem Beitrag beschriebene System wurde im Auftrag der Mitsubishi Corporation am Fraunhofer-IGD, Darmstadt, entwickelt. Eine Produktversion ist in Form des ReEncryption-Systems für Betriebssysteme der Microsoft Windows NT-Familie (Windows 2000) verfügbar.

Für dieses System existieren weitere Komponenten als realisierte Prototypen, welche verbesserte Sicherheitsmerkmale bei Bedarf unterstützen (z.B. für Mehrfaktor-I&A-Mechanismen in Form von SmartCards oder durch Hardware geschützte Coprozessoren) können sowie Integrationsmechanismen wie die Synchronisation von Nutzerdaten mit bestehenden über LDAP (Lightweight Directory Access Protocol (Wahl, Howes und Kille 1997)) ansprechbaren Nutzerdatenbanken.

Darüber hinaus existieren Prototypen für Unix-Systeme, insbesondere für kommerzielle System V Release 4-Derivate (Sun Solaris 8, 32 und 64 Bit Kernel), welche voll mit den auf Microsoft Windows basierenden Komponenten interoperabel sind und einer Produktversion zugeführt werden.

Weitere mögliche Ergänzungen sind z.B. in der engen Integration mit bestehenden Dokumenten-Management- und Workflow-Management-Systemen realisierbar; dadurch lassen sich die hier diskutierten Sicherheitsmerkmale transparent in bestehende Arbeitsabläufe und Datenhaltungen integrieren.

Literatur

Arnold, M., Schmucker, M. und Wolthusen, S. D.: 2003, *Digital Watermarking and Content Protection: Techniques and Applications*, The Artech House Computer Security Series, Artech House, Norwood, MA, USA.

Bertino, E. und Ferrari, E.: 2002, Secure and Selective Dissemination of XML Documents, *ACM Transactions on Information and System Security* **5**(3), 290–331.

Blaze, M.: 1993, A Cryptographic Filesystem for Unix, *Proceedings of the 1st ACM Conference on Computer and Communications Security*, ACM Press, Fairfax, VA, USA, pp. 9–16.

Busch, C. und Wolthusen, S.: 2001, Tracing Data Diffusion in Industrial Research with Robust Watermarking, *in* J.-L. Dugelay und K. Rose (Hrsg.), *Proceedings of the 2001 Fourth Workshop on Multimedia Signal Processing (MMSP'01)*, IEEE Press, Cannes, Frankreich, pp. 207–212.

Busch, C. und Wolthusen, S. D.: 2002, *Netzwerksicherheit*, Spektrum Akademischer Verlag, Heidelberg.

Cox, I. J., Miller, M. L. und Bloom, J. A.: 2002, *Digital Watermarking*, The Morgan Kaufmann Series in Multimedia Information and Systems, Morgan Kaufmann Publishers, San Francisco, CA, USA.

Dierks, T. und Allen, C.: 1999, RFC 2246: The TLS Protocol Version 1.0, Internet Engineering Task Force Request For Comments.

Dourish, P., Edwards, W. K., LaMarca, A., Lamping, J., Petersen, K., Salisbury, M., Terry, D. B. und Thornton, J.: 2000, Extending Document Management Systems with User-specific Active Properties, *ACM Transactions on Information Systems* **18**(2), 140–170.

Gifford, D. K., Jouvelot, P., Sheldon, M. A. und O'Toole, Jr., J. W.: 1991, Semantic File Systems, *ACM Operating Systems Review* **25**(5), 16–25. Proceedings of the Thirteenth ACM Symposium on Operating Systems Principles.

Harkins, D. und Carrel, D.: 1998, RFC 2409: The Internet Key Exchange (IKE), Internet Engineering Task Force Request For Comments.

Mauriello, E.: 1997, TCFS: Transparent Cryptographic File System, *Linux Journal* **40**.

McCoy, K.: 1990, *VMS File System Internals*, Butterworth-Heinemann, Maynard, MA, USA.

Rademer, E. und Wolthusen, S.: 2001, Transparent Access To Encrypted Data Using Operating System Network Stack Extensions, *in* R. Steinmetz, J. Dittman und M. Steinebach (Hrsg.), *Communications and Multimedia Security Issues of the New Century: Proceedings of the IFIP TC6/TC11 Fifth Joint Working Conference on Communications and Multimedia Security (CMS'01)*, IFIP, Kluwer Academic Publishers, Darmstadt, pp. 213–226.

Thayer, R., Doraswamy, N. und Glenn, R.: 1998, RFC 2411: IP Security Document Roadmap, Internet Engineering Task Force Request For Comments.

Wahl, M., Howes, T. und Kille, S.: 1997, RFC 2251: Lightweight Directory Access Protocol (v3), Internet Engineering Task Force Request For Comments.

Whitten, A. und Tygar, J. D.: 1998, Usability of Security: A Case Study, *Technical Report CMU-CS-98-155*, Carnegie Mellon School of Computer Science, Pittsburgh, PA, USA.

Zadok, E.: 2001, *FiST: A System for Stackable File System Code Generation*, Dissertation, Columbia University, New York, NY, USA.