# NOTITIA

AdNovum

BEMERKENSWERTES VON UND ÜBER ADNOVUM

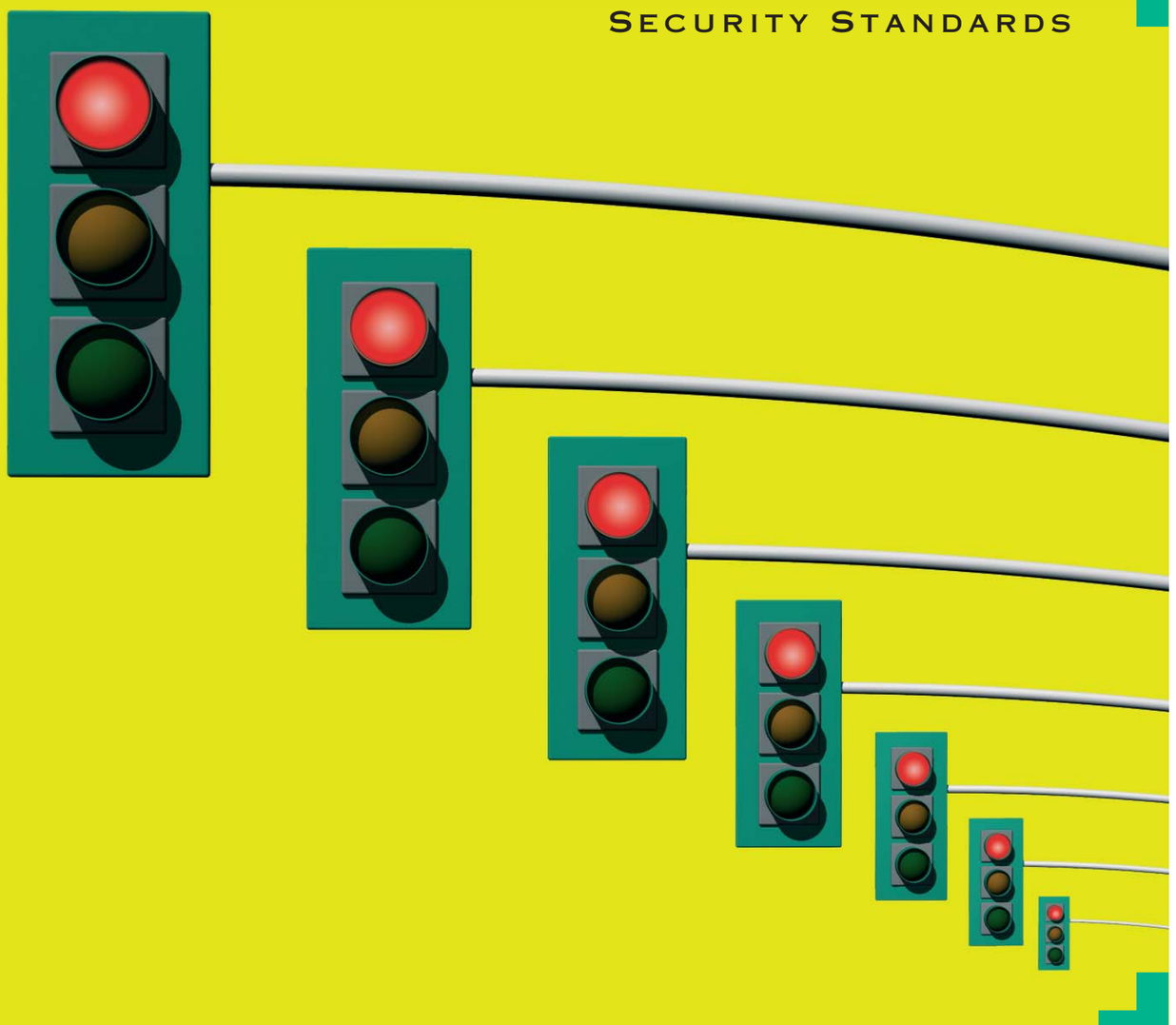
## Einsatz von Security Standards zur Trust-Etablierung
Hohe Sicherheit und Herstellerunabhängigkeit

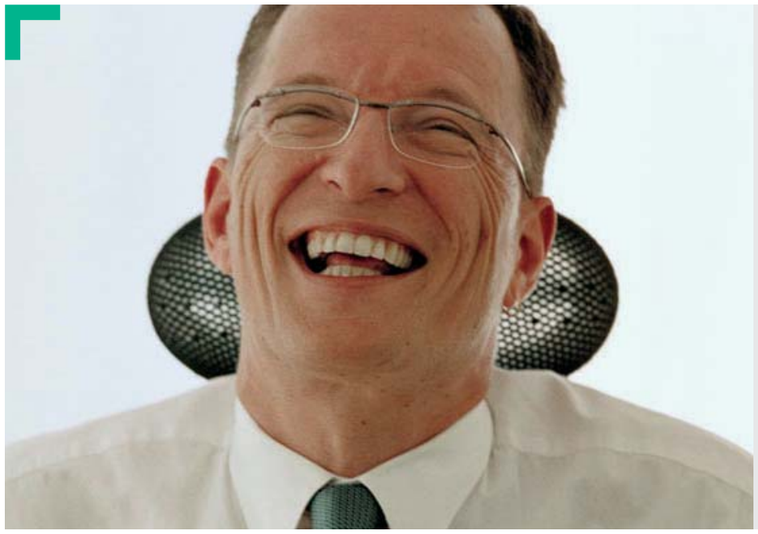
## Secure Identity Management
Strategien, Prozesse, Technologie, Technik

## Multiprotokollfähiges Portal
Single-Signon in einem heterogenen Umfeld

HERBST 2004, NR. 7

SECURITY STANDARDS





Liebe Leserin, lieber Leser

Nein, Sie haben kein Déjà-vu-Erlebnis, auch in dieser Ausgabe der Notitia steht im Haupttitel «Security». Das soll aber nicht heissen, dass wir Ihnen etwas Aufgewärmtes vorsetzen wollen, denn es gibt zweifellos noch genug Neues zu diesem Thema zu sagen. Die ständig zunehmende Vernetzung und starke Förderung des verteilten Zugriffs auch auf sicherheitskritische Daten machen die Absicherung der Systeme und Kommunikationswege unabdingbar. Darum sind wir überzeugt, dass Sicherheit im IT-Bereich nach wie vor ein höchst aktuelles Thema ist.

In letzter Zeit hat sich der Einsatzbereich der Lösungen erweitert. War Security früher ein Thema, das fast ausschliesslich die Bankenwelt und die Wirtschaft interessierte, geniesst es nun auch andernorts Aufmerksamkeit. Dies ist sicher nicht zuletzt den Kostensenkungen zu verdanken, die in diesem Bereich möglich wurden. Solange Sicherheitslösungen vorwiegend im Umfeld von Finanzdienstleistern eingesetzt wurden, die von der Einhaltung höchster Sicherheitsvorschriften besonders abhängig sind, spielten Kosten vermeintlich eine untergeordnete Rolle. Dies hatte zur Folge, dass lange Zeit keine Standards entwickelt und eingesetzt wurden, die verwendeten Basistechnologien waren teure Individuallösungen. Dadurch entstanden aber nicht nur bei der Entwicklung hohe Kosten, sondern auch im Betrieb, denn die Lösungen wiesen komplizierte und aufwändige Betriebsmuster auf. Mittlerweile wurden die Standards so weit entwickelt, dass sich auch Sicherheitslösungen, an die hohe Anforderungen gestellt werden, mit einem durchaus vertretbaren Mitteleinsatz realisieren lassen.
In dieser Ausgabe der Notitia möchten wir Ihnen einige dieser etablierten Standards vorstellen, wie zum Beispiel die Trust-Etablierung, welcher der Hauptartikel gewidmet ist. Am Beispiel Identity Management zeigt sich, wie heute dank Komponenten für den Aufbau einer Sicherheitsinfrastruktur Lösungen mit modernen Mitteln effizient entwickelt und betrieben werden können.

Stefan Arn

CEO AdNovum Informatik AG

# Einsatz von Security Standards zur Trust-Etablierung

Sicherheit in verteilten Systemen beruht in vielen Fällen auf Vertrauen in andere Instanzen, das nicht immer begründet ist. Damit die oft über mehrere Knoten propagierten kritischen Informationen vertrauenswürdig sind, müssen spezielle Mechanismen eingesetzt werden. Dabei wird mit Vorteil auf Standards zurückgegriffen, da diese mehr Sicherheit und Herstellerunabhängigkeit bieten.

Von Philipp Färber

Vertrauen ist gut, birgt aber immer auch Risiken – vor allem wenn es sich nicht gut begründen lässt. Zum Beispiel ist das Vertrauen in den Compiler oder die Laufzeitumgebung des Betriebssystems normalerweise ohne Code Review gerechtfertigt. Bei anderer «fremder» Software kann dagegen eine Überwachung des Laufzeitverhaltens oder die Verifikation einer Code-Signatur durchaus nötig sein: Kontrolle ist eben besser.

Ein aktueller Bereich, in dem leider allzu oft noch «blindes» Vertrauen herrscht, ist die Kommunikation zwischen verschiedenen Software-Komponenten. Zwar hat sich inzwischen die Ansicht durchgesetzt, dass Kommunikationspartnern im Internet grundsätzlich misstraut werden muss, – im (vermeintlich sicheren) Intranet wird aber häufig ganz auf die Verbindungssicherung verzichtet. Dabei steht hier mit dem SSL-Protokoll zusammen mit dem X.509-Zertifikats-Standard ein bewährter Mechanismus zur Verfügung, mit dem sich beide Kommunikationspartner mit hoher Güte (das heisst «stark») gegenseitig authentisieren können. Das Vertrauen in die Identität des Peers wird dabei durch dessen digitale Signatur zum Zeitpunkt des Handshake gerechtfertigt.

Leider beschränkt sich eine derart etablierte gegenseitige Authentisierung auf Basis von SSL immer auf die beiden Peers, wir sprechen deshalb auch von Point-to-Point Security. Typischerweise erstreckt sich die Kommunikation jedoch über verschiedene Zwischenstationen, seien dies technische Komponenten wie Proxies und Gateways oder applikatorische Tiers (vergleiche Abbildung mit einer Multi-Tier-Architektur mit Reverse Proxy). Wie kann die Identitätsinformation vertrauenswürdig an einen entfernten Peer weitergegeben werden? Umgekehrt gefragt: Wie kann der Server S3 sein Vertrauen in die Identität des Client C rechtfertigen? Eine direkte Point-to-Point-Authentisierung analog zu SSL ist bezüglich Skalierbarkeit, technischer Machbarkeit und Ergonomie für den Benutzer nicht praktikabel.

## Secure Delegation statt blindes Vertrauen

Ein verbreiteter Umgang mit dem vorliegenden Problem besteht darin, das Vertrauen in den Nachbarn transitiv anzuwenden: So vertraut der Server S2 z. B. den Informationen von S1, da er S1 direkt authentisiert hat, welcher wiederum dem Reverse Proxy vertraut, der den Client authentisiert hat. Obwohl diese Folgerung einleuchtet, muss man sich bewusst sein, dass dadurch das Vertrauen implizit auf alle vorangegangenen Server ausgedehnt wird – oft ohne diese überhaupt zu kennen. Ein allfälliger Fehler bei der Authentisierung wird in diesem Fall blind weiterpropagiert.

Zusätzlich zur Authentisierung des Peers erhöht man deshalb die Vertrauenswürdigkeit der propagierten Informationen, indem man diese von einer zentralen und besonders vertrauensanker, als so genannte «Trusted CA» gesicherten Instanz signieren lässt. Die Signierinstanz übernimmt dabei eine ähnliche Rolle wie eine klassische Certificate Authority, die bei der Zertifikatsvalidierung (zum Beispiel beim SSL-Verbindungsaufbau) explizit als Vertrauensanker, als so genannte «Trusted CA» dient. Werden die signierten Daten (das heisst das Token) zusätzlich noch mit einer relativ kurzen Lebensdauer (Session Lifetime oder kürzer) versehen, bietet sich auch ein gewisser Schutz gegen einfache Replay Attacken, bei denen ein altes Token zu einem späteren Zeitpunkt missbraucht wird.

Die signierten Tokens haben noch einen weiteren Vorteil: Da die Signatur im Gegensatz zu SSL nicht online erfolgt, kann ein Token auch bei asynchronen Kommunikationskanälen benutzt werden.

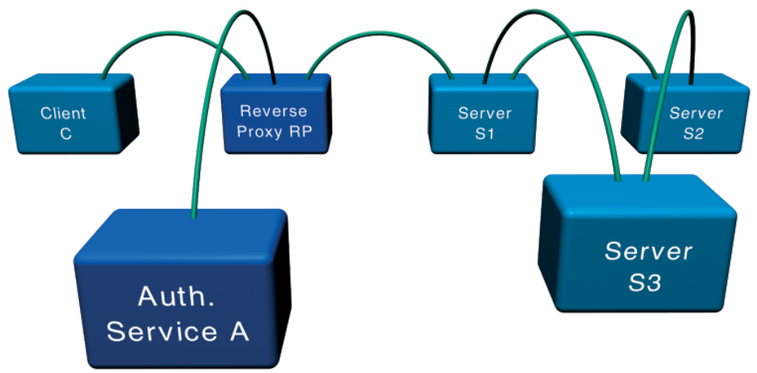
Die beschriebenen Mechanismen erlauben die Realisierung eines verteilten Systems mit Secure Delegation, in dem Vertrauen auch gegenüber entfernten Peers begründet ist. In der Nevis-Web-Architektur übernimmt der Authentisierungsservice die Rolle der zentralen Signierinstanz für Identitätsinformationen, indem er nach erfolgreicher Authentisierung des Benutzers ein so genanntes Secure Token erstellt. Neben der User-Identifikation kann dieses Token auch Berechtigungen oder Rollen-Information sicher transportieren.

## Standards sind besser …

Bei der Realisierung dieser Ansätze stösst man sehr schnell auf die Frage, über welchen Mechanismus die kritische Information propagiert werden soll, ohne dass die Lösung herstellerabhängig oder unsicher wird – gerade wenn es darum geht, viele «fremde» Komponenten sicher zu integrieren.

> Die zentrale Signierinstanz für sichere Tokens entspricht der klassischen «Certificate Authority» für X.509-Zertifikate.

Die Erfahrung der AdNovum hat gezeigt, dass sich die konsequente Nutzung offener Standards bewährt, denn diese bieten hohe Sicherheit durch ein öffentliches Review-Verfahren sowie Unterstützung durch viele Hersteller und damit (zumindest in der Theorie) Interoperabilität zwischen verschiedenen Produkten. Zur Realisierung werden zwei grundsätzliche Funktionen benötigt, die am besten im Bereich der Middleware realisiert sind. Zum einen braucht es ein Transportgefäss innerhalb eines Request in einem so genannten Context oder Envelope und zum anderen den applikatorischen Zugriff auf die propagierte Benutzerinformation über ein API. Auch hier bietet die Einbettung in die Landschaft etablierter Standard eine verbesserte Sicherheit und eine transparente Integration.




*Beispiel einer verteilten Multi-Tier-Architektur mit Reverse Proxy.*

*Philipp Färber unterstützt mit seinem Security Know-how diverse Projekte.*

Als konkretes Fallbeispiel soll im Folgenden die Weitergabe der authentisierten Benutzeridentität (Principal) durch verschiedene J2EE Container dienen. Hier akzeptieren die gängigen Applikationsserver neben proprietären Mechanismen vermehrt auch den CSIv2 Standard der OMG (siehe Kasten), der zumindest im Level 0 für EJB Container vorgeschrieben ist. Da sich die darin definierten Mechanismen in der Protokollebene GIOP abspielen, sind damit aber nicht nur Java, sondern auch allgemeine CORBA Services integrierbar.

> Die konsequente Nutzung offener Sicherheits-Standards bietet gleichzeitig ein hohes Sicherheitsniveau und hohe Interoperabilität.

Wie findet die Weitergabe der Principal-Information über CSIv2 statt? Im Gegensatz zu SSL läuft bei CSIv2 kein mehrstufiger Handshake ab, bei dem Sicherheitsparameter ausgehandelt und Signaturen ausgetauscht werden, sondern die zugesicherte Information befindet sich bereits im ersten Paket des Request zum Server. Dieser entscheidet anhand der Qualitätsparameter der Transportschicht (also der SSL-Verbindung), ob der Request seinen Sicherheitsanforderungen entspricht und beurteilt dann die darin enthaltene Authentisierungsinformation. Die Sicherheitsanforderungen sind für jeden Service fest vorgegeben und werden über dessen IOR (Interoperable Object Reference) publiziert.

Das Vertrauen in die Identität des Aufrufers wird dabei aus der Authentisierung der SSL-Schicht gewissermassen «geerbt». Es kann vorkommen, dass der Client kein Zertifikat besitzt (oder keine direkte SSL-Verbindung besteht), in diesem Fall kann der Name des Principal optional auch mittels eines mitgeschickten Shared Secret (zum Beispiel Passwort) verifiziert werden. Dies entspricht dem CSIv2 Conformance Level 0.

> Leicht erweiterbare Standards wie SAML sind zwar flexibel, führen aber leicht zu herstellerabhängigen Lösungen.

Höhere Conformance Levels benutzen zusätzlich X.509 Attribute Certificates, mit denen eine wohldefinierte Secure Delegation realisiert werden kann. Das Vertrauen in die Principal-Informationen wird dabei durch signierte Attribut-Zertifikate erhöht, die den Zwischenstationen explizit die Erlaubnis erteilen, für den Principal als Proxy zu fungieren.

In der Nevis-Web-Architektur wird CSIv2 eingesetzt, um die signierten Secure Tokens weiterzugeben – allerdings kann man derzeit erst die Mechanismen von Conformance Level 0 einsetzen, da die aktuellen (J2EE Application) Server noch keine höheren Levels unterstützen.

### «End 2 End» (... gut, alles gut?)

Als weiterer Standard zur Weitergabe vertraulicher Informationen auf der Service-Ebene sei hier noch die Security Assertions Markup Language (SAML) von OASIS erwähnt. SAML definiert eine XML-Syntax zur signierten Weitergabe von sicherheitsrelevanten Aussagen (etwa «Client C wurde um 13:27 von Server RP mittels Zertifikat authentisiert»), womit sich z. B. Single-Signon-Lösungen für stark heterogene Komponenten realisieren lassen.

In Kombination mit den beiden W3C Standards XML-DSIG und XML-ENC, die digitale Signaturen und Verschlüsselung auch über einzelne Teile von XML-Dokumenten erlauben, bietet SAML ein flexibles Format für vertrauliche Informationen. Die oben geschilderten Varianten «Point-to-Point Security» (durch direkte Signatur des Absenders) und «Secure Delegation mit signierten Token» lassen sich in SAML sogar kombinieren, da ein SAML-Fragment aus mehreren unabhängigen Teilen bestehen kann. So kann eine SAML Assertion mehrere Zusicherungen enthalten, die an verschiedene Zielserver adressiert sind (z. B. eine vom Kunden signierte Bestellung und eine von der Bank signierte Kreditzusage).

Ganz im Sinne allgemeiner Web-Services lassen sich so vertrauenswürdige Daten unabhängig von den Eigenschaften der Transportschicht kommunizieren – obige SAML Assertion liesse sich ohne weiteres per Mail verschicken.

SAML ist sehr offen und erweiterbar spezifiziert. Dies ist zwar gut im Hinblick auf die Flexibilität, wirkt sich aber negativ auf die Interoperabilität aus, da jeder Hersteller seine Security Assertions anders definieren oder auslegen kann. In der Nevis-Web-Architektur wird SAML eingesetzt, um einen Single-Signon-Verbund zwischen ansonsten unabhängigen, bereits existierenden Nevis-Web-Instanzen herzustellen.

### Fazit

Zusammenfassend lässt sich sagen, dass wohl jedes realistische IT-System gewissen Informationen vertrauen muss, dieses Vertrauen aber unbedingt begründet sein sollte. Insbesondere im Bereich der sicheren Kommunikation kann dieses Ziel mit Hilfe von digitalen Signaturen erreicht werden. Im Sinne einer hohen Sicherheit und einer breiten Interoperabilität sollte die Umsetzung wenn möglich durch offene Standards erfolgen. In unseren Lösungen haben wir dabei mit den hier vorgestellten Standards SSL, X.509, CSIv2 und SAML durchwegs positive Erfahrungen gemacht. ■

**Philipp Färber**

*Philipp Färber studierte in München, Colorado und an der ETH Zürich Elektrotechnik und bringt in der AdNovum seit etwa einem Jahr sein Know-how im Bereich Security in diverse Projekte ein. Daneben versucht er ausdauernd (aber bisher relativ erfolglos), seine Kollegen zu gemeinsamen Marathonläufen zu motivieren.*

### CSIv2

Die Spezifikation Common Secure Interoperability Version 2 ist von der Object Management Group (OMG) als integraler Bestandteil in die CORBA-Core-Spezifikation Version 2.6 aufgenommen worden und damit auch zu einem Bestandteil von J2EE geworden. Sie beschreibt ein Message-Protokoll CORBA Security Attribute Service (SAS), welches auf der Basis von regulären CORBA Request und Reply Messages funktioniert und einen sicheren Transport (typischerweise SSL) voraussetzt. Dazu definiert SAS den Authentication und den Attribute Layer oberhalb des Secure Transports. Die CSIv2 Messages werden in einem speziellen Service Context des CORBA Protokolls GIOP transportiert.

Mit dem Authentication Layer lassen sich sowohl Authentisierungsinformationen (GSS Tokens) als auch delegierte Identitäten (GSS Principal, X.501 Distinguished Names oder X.509 Certificate Chains) für so genannte Identity Assertions auf dem Server transportieren.

Auf dem Attribute Layer können zusätzlich Autorisierungs-Tokens, welche von einer Drittinstanz ausgestellt wurden, vom Client zum Server transportiert werden. Aufgrund eines solchen Autorisierungs-Token kann der Server zum Beispiel entscheiden, ob eine Secure Delegation überhaupt zulässig ist.

Der CSIv2 Standard definiert drei Levels von Conformance, wobei das unterste Niveau, Level 0, nur den Authentisierungs-Layer beinhaltet. Level 1 und 2 verlangen die Unterstützung des Autorisierungs-Token in zwei Stufen. Der aktuelle J2EE Standard verlangt für die Unterstützung von sicheren EJB Service Calls über RMI/IIOPS die Implementierung von CSIv2 Level 0.

# Secure Identity Management

Christian Grob sprach mit der Notitia über Grundideen des aktuellen Themas Identity Management und die technischen und betriebsorganisatorischen Prozessanpassungen, die für eine unternehmensweite Umsetzung der globalen Benutzeridentität nötig sind.

Interview: Barbara Stammler

*NOTITIA:* Ist Identity Management eine neue Technologie oder ein neues Konzept?

*Christian Grob:* Weder noch, einige Grundideen des Identity Managements (IDM) decken sich mit bekannten Konzepten. So haben Benutzerdaten und Identitäten im Allgemeinen für Informatik-Systeme immer eine zentrale Rolle gespielt. CRM-Lösungen (Customer Relationship Management) beispielsweise kann man als IDM-Systeme im Kleinen ansehen, besteht doch eines ihrer Ziele darin, für das ganze Unternehmen eine einheitliche Sicht des Kunden und der Beziehungen zu ihm zur Verfügung zu stellen. Diese unternehmensweite Sicht einer Identität ist auch ein Hauptanliegen des IDM. Neu ist die konsequente Verwendung einer einzigen Identität in allen Systemen einer Unternehmung.

Ein anderes Beispiel dafür, dass IDM keine neue Idee ist, sind die heute weit verbreiteten Single-Signon-Portale. Diese Technologie ermöglicht nach einmaliger Authentisierung den Zugriff auf unterschiedliche und unabhängige Applikationen.

> « IDM ist auf Prozessoptimierung und Organisation ausgerichtet mit dem Ziel, die Effizienz zu steigern und die Kosten zu senken. »

Sind IDM-Aspekte nicht bereits genügend abgedeckt und gelöst, oder anders gefragt, ist IDM alter Wein in neuen Schläuchen?

Das IDM stellt andere Ansprüche an dasselbe Thema. Im Zuge der Dezentralisierung wurden Identitäten mit ihren Authentisierungs- und Autorisierungsmerkmalen repliziert und über die Jahre in eigener Verantwortung des jeweiligen Systems weitergepflegt. Das hat dazu geführt, dass in Grossfirmen jeder Mitarbeiter heute oft zahlreiche autorisierte Accounts besitzt. Einerseits sind die Benutzer damit überfordert, andererseits entstehen Kosten durch vergessene Passwörter und verlorene Zugangskarten. Werden die Systeme zur Datenpflege von Identitäten, Authentisierungsmerkmalen und Autorisierungsregeln für jedes Projekt neu bereitgestellt, fallen zusätzlich Entwicklungs-, Konfigurations- und Betriebskosten an. Diese akkumulierten Aufwendungen können heute nicht mehr vernachlässigt werden.

Mit dem Paradigma der Dezentralisierung wurde während des Internetbooms den Folgen der Replizierung von Daten und Funktionalität wenig Beachtung geschenkt. Mit zunehmendem Kostendruck müssen nun Prozessoptimierungen durchgeführt werden, um die Effizienz zu steigern und die Kosten zu senken. Darin besteht im Prinzip die neue Sicht des IDM: Sie ist auf Prozessoptimierung und Organisation ausgerichtet. Identity Management ist die konsequente Weiterführung der Ideen aus CRM und SSO (Single Signon).

Um auf die Metapher zurückzukommen: Die alten Weinschläuche sind weit verzweigt und zu einem unüberschaubaren Knäuel angewachsen. Zudem sind die Schläuche rissig und verlieren den Wein.

Welche Informationen gehen denn verloren?

Unabhängig davon, ob wir nun Identitäten von Kunden oder Mitarbeitern betrachten, müssen wir feststellen, dass durch die Entkopplung und Redundanz der Repositories wertvolle Informationen verloren gehen wie z. B. Veränderungen in Kundenbeziehungen. Gravierender sind die Folgen, wenn beim Austritt eines Mitarbeiters diese Information zwar im Personalsystem vorhanden ist, in den dezentralen Repositories aber nicht weiterverwertet wird. So bleiben Authentisierungs- und Autorisierungsdaten länger als notwendig gültig, was der Sicherheit natürlich abträglich ist.

Die Security leidet aber nicht nur in dieser Hinsicht, sondern auch dadurch, dass Kontrollen kaum mehr durchführbar sind. Denn die dezentrale Datenhaltung der Rechte ohne unternehmensweites Konzept erschwert oder verunmöglicht deren Management.

Darüber hinaus schreibt das Gesetz für viele Unternehmen sogar eine Nachweispflicht für Autorisierungen vor. Identity Management verbessert hier die Transparenz und Nachvollziehbarkeit entscheidend. Steht die Sicherheit im Vordergrund, sprechen wir von Secure Identity Management (SIDM). Interessant ist, dass die Realisierung von SIDM zusammen mit einem SSO-Portal doppelten Gewinn bringt. Sie wirkt sich sowohl positiv auf den Benutzerkomfort wie auch auf die Sicherheit aus. Das ist selten, wird doch in der Regel mehr Sicherheit mit einem Verlust an Benutzerkomfort bezahlt.

> « Die dezentrale Datenhaltung der Rechte ohne unternehmensweites Konzept erschwert oder verunmöglicht deren Management. »

Für die Umsetzung dieser Anforderungen sind mittlerweile viele IDM-Produkte auf dem Markt. Wie wird aus diesen Produkten ein IDM-System aufgebaut?

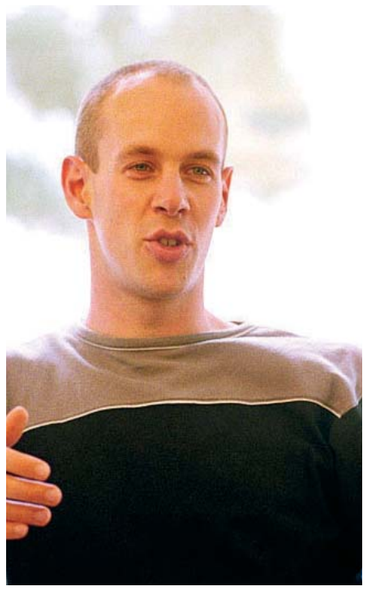
Ein IDM-System besteht aus mehreren Komponenten. Wichtig ist die zentrale Datenhaltung, bei der ein Directory, aber auch relationale Datenbanken zum Einsatz kommen. Die Daten werden direkt oder über Services mit zusätzlicher Funktionalität angeboten, zum Beispiel einen Authentisierungsservice. Daneben sind auch Synchronisations-Mechanismen mit Umsystemen notwendig.

Nicht zuletzt wollen die Identitäten gepflegt werden, einerseits durch die Benutzer selbst (ihre eigenen Daten), andererseits durch Administratoren in verschiedenen Rollen. Die unternehmensweiten Identitätsattribute wie Personendaten oder Grobautorisierungen werden typischerweise in einer zentralen Applikation gepflegt. Spezifische Applikationsdaten, verknüpft mit der Identität, werden in den einzelnen Applikationen verwaltet.

Dies zeigt, dass ein einzelnes Produkt nicht alle Aufgaben eines IDM-Systems abdecken kann, ein solches System besteht darum aus verschiedenen aufeinander abgestimmten Komponenten. Die IDM-Produkte auf dem Markt bieten wertvolle Bausteine für den Aufbau eines solchen Systems. Neben den Standard-Komponenten wie Datenbanken und Directories verkürzen Tools zur Synchronisation und Datenpflege die Entwicklungszeit.

Können die marktgängigen Produkte alle Anforderungen abdecken?

Beim Design eines IDM in Zusammenarbeit mit einem Kunden haben wir die Erfahrung gemacht, dass, sobald man etwas in die Tiefe vordringt, sehr schnell unternehmensspezifische Anforderungen aufkommen. Die Regelwerke eines Produktes können nur in den wenigsten Fällen alle Aspekte abdecken. Grundlegende Eigenheiten wie die Abbildung



spezifischer organisatorischer Prozesse erfordern Erweiterungen des IDM-Produktes. Sorgfalt bei der Auswahl eines Produkts zahlt sich hier auf jeden Fall aus, insbesondere wenn bestehende Applikationen integriert werden müssen. Ein spezielles Augenmerk richten wir auf den geforderten Security Level und wie weit dieser durch das Produkt abgedeckt ist.

**Worauf muss bei der Einführung einer SIDM-Lösung geachtet werden?**

Der Aufbau eines IDM-Systems findet praktisch nie auf der grünen Wiese statt. Im Gegenteil, meist müssen mehrere Benutzerstämme konsolidiert werden, die nicht kompatibel sind, was an und für sich schon mit einem grossen Aufwand verbunden ist. Auch die Authentisierungsdaten der am SIDM teilnehmenden Applikationen müssen auf einen Nenner gebracht werden. Dieser Prozess verlangt ein unternehmensweites Datenmodell über Benutzer, Applikationen, Systeme, Ressourcen, Rollen und Berechtigungen. Diese einheitliche Sicht vereinfacht die Informatik-Umgebung und fördert damit das Verständnis für Daten und Abläufe. Auf dieser Basis kann durch technische und organisatorische Massnahmen die Sicherheit erhöht werden.

Vorgehen und Projektmanagement sind bei der Einführung eines SIDM von besonderer Bedeutung. In Diskussionen über Benutzer, Berechtigungen, Datenraumautorisierung und ähnliche Begriffe aus dem Umfeld des SIDM treten schnell Missverständnisse auf. Nicht selten muss eine gemeinsame Sprache gefunden werden.

**Gibt es Fälle, in denen man von der Realisierung einer SIDM-Lösung abraten muss?**

Eine SIDM-Lösung macht vor allem dann Sinn, wenn nach dem Pareto-Prinzip (80/20-Regel) die strategischen Applikationen integriert werden. Für Applikationen mit nur noch kurzer Lebenszeit geht die Kosten-Nutzen-Rechnung bei einer Integration in das SIDM meist nicht auf. Im Fall von mehreren Projekten mit langer Laufzeit tritt das einzelne IDM-Produkt in den Hintergrund. Hier steht das Überwinden von Abteilungsdenken bei der Zusammenarbeit im Vordergrund, dabei kann ein neutraler Integrator oft wertvolle Hilfe leisten.

Damit die verschiedenen Applikationsverantwortlichen in der SIDM-Lösung nicht nur einen Mehraufwand sehen, muss ein Mehrwert für die Anwendungen entstehen. Dieser kann eine ausgelagerte und einfache Authentisierung, Autorisierung, Tooling und Ähnliches sein. Den Bedenken gegen die Anbindung an ein vermeintlich träges System steht die Gefahr gegenüber, dass jede Applikation ihre eigene Lösung für Secure Identity Management umsetzt und betreibt, was dem Ziel des SIDM widerspricht. Der Entscheid für eine unternehmensweite SIDM-Lösung ist damit von strategischer Natur. ■

> «Der Entscheid für eine unternehmensweite Lösung für das Secure Identity Management ist von strategischer Natur.»

**Christian Grob**

*Christian Grob, ETH-Software-Ingenieur, arbeitet seit 1999 in der AdNovum. Er beschäftigt sich im Umfeld von Applikationen mit deren Middleware- und Software-Engineering-Aspekten. Neue Technologien stehen dabei auf der Tagesordnung. Er prüft sie auf ihren Nutzen und Tauglichkeit für den produktiven Einsatz und arbeitet bei der Umsetzung in den Projekten mit. Zum Ausgleich lüftet er seine Gedanken mit Mountain-Biken im Jura-Gebirgszug aus.*



# Multiprotokoll-fähiges SSO-Portal

Ein neues Single-Signon-Portal ermöglicht den einheitlichen Zugriff auf die Netzwerke und Anwendungen des Eidgenössischen Justiz- und Polizeidepartements (EJPD). Da mit besonders sensiblen Personendaten gearbeitet wird, muss das Portal höchste Sicherheitsanforderungen erfüllen. Die Lösung integriert auf der Basis der Nevis-Web-Architektur ganz unterschiedliche Systeme: Web-Applikationen (zum Teil auf der Basis von J2EE), Host-Applikationen (Terminal-Emulationen) und Rich Clients mit proprietären Protokollen.

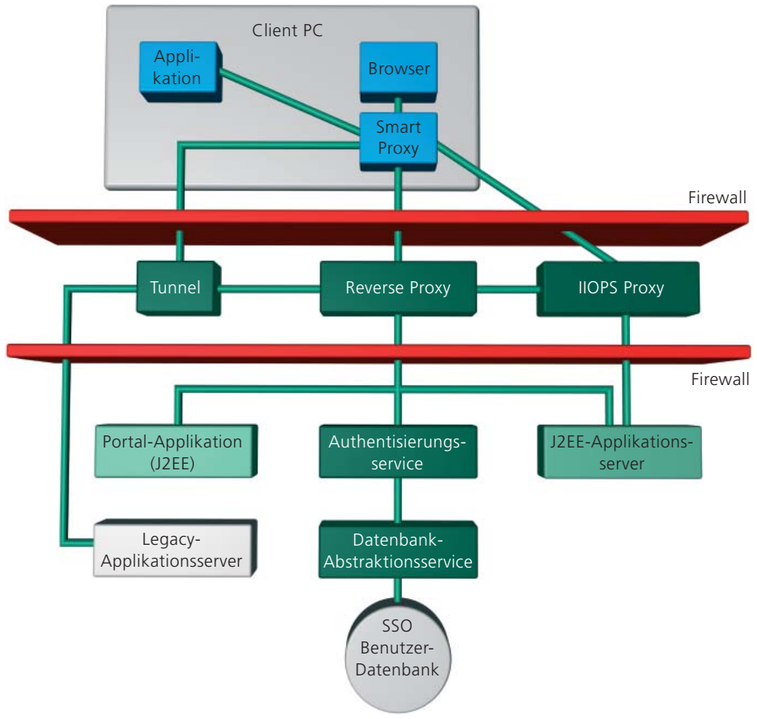
Von Andreas Signer

Die Anwendungslandschaft des Eidgenössischen Justiz- und Polizeidepartements ist wie in vielen anderen Grossbetrieben über die Jahre gewachsen und damit sehr heterogen. Verschiedene Anwendungstypen und -generationen, mit teilweise eigenen Benutzerverwaltungen und Authentisierungsinfrastrukturen existieren nebeneinander. In vielen dieser Applikationen werden zudem proprietäre Protokolle verwendet. Das neue Single-Signon-Portal (SSO-Portal) platziert alle Anwendungen in einer logischen Sicherheitsschicht und bietet eine einheitliche Lösung für Authentisierung und Autorisierung. Authentisierung und Grobautorisierung – das heisst die Kontrolle darüber, ob ein Benutzer überhaupt auf eine bestimmte Anwendung zugreifen darf – werden neu von einem zentralen Service erledigt. Die Feinautorisierung, bei der die Berechtigung zur Verwendung bestimmter Funktionen und der Zugriff auf die Daten innerhalb einer Applikation geregelt werden, erfolgt weiterhin in den einzelnen Applikationen.

Aufgrund der vielen verschiedenen und teilweise proprietären Protokolle, die gleichzeitig eingesetzt werden, musste eine multiprotokollfähige Lösung entwickelt werden. Während übliche SSO-Portale über eine Integrationsschicht Applikationen meist nur in Form von spezifischen Web-Anwendungen im Browser nutzbar machen, ist es im SSO-Portal des EJPD auch möglich, vom Portal aus Programme und Daten in ihrer ursprünglichen Form aufzurufen und damit zu arbeiten. Das heisst, es werden statt Inhalten Kommunikationskanäle konsolidiert, indem Applikationen und Programme harmonisiert werden.

## Technische Umsetzung

Das SSO-Portal implementiert die Zugriffsschicht (Access Layer) der neuen sicheren J2EE-Anwendungsarchitektur des EJPD. Alle Applikationen sind in einer Sicherheitszone untergebracht, dem so genannten Application Backbone, auf den nur über das SSO-Portal zugegriffen werden kann. Um den hohen Anforderungen an die Sicherheit gerecht zu werden, erfolgt die Kommunikation innerhalb des Single-Signon-Verbundes über sämtliche Netzwerkstrecken verschlüsselt. Alle an der Kommunikation beteiligten Komponenten können anhand von Zertifikaten jederzeit eindeutig identifiziert werden. Die dazu notwendige Sicherheits- und Authentisierungsinfrastruktur wurde auf der Basis der modular aufgebauten Security-Integrationsplattform Nevis realisiert.




*Aufbau des Single-Signon-Portals.*

## Standard-Nevis-Komponenten

Das Portal ist auf der Basis der Standard-Nevis-Komponenten Reverse Proxy, Authentisierungs- und Datenbankabstraktions-Service aufgebaut, zu denen speziell für das EJPD-Portal entwickelte Komponenten kommen.

Der Secure Reverse Proxy dient als einheitlicher Einstiegspunkt für alle integrierten Anwendungen, als so genannter Single Point of Entry. Er überprüft jeden Benutzerzugriff und stellt in Zusammenarbeit mit dem Authentisierungsservice sicher, dass nur entsprechend authentisierte und autorisierte Benutzer auf die Anwendungen zugreifen können.

*Andreas Signer war als technischer Projektleiter für die Realisierung des SSO-Portals verantwortlich.*

Der Authentisierungsservice ist für die Authentisierung der Portalbenutzer und die Verwaltung der Benutzerdaten sowie der User Sessions im Portal zuständig. Sein modularer Aufbau erlaubt es, verschiedene Authentisierungsmechanismen flexibel zu integrieren, wie zum Beispiel User ID/Passwort, Secure ID, Challenge/Response und x.509v3-Zertifikate. Dies ermöglicht auch eine Migration auf neue Technologien, ohne die bestehende Anwendungslandschaft zu beeinträchtigen. Die Authentisierungsdaten werden in einer separaten Datenbank verwaltet. Mit der Auslagerung der Authentisierung in einen dedizierten Service ist die Basis für zukünftige Authentisierungsmechanismen gelegt.

Nach der erfolgreichen Authentisierung werden die Benutzerattribute in ein Secure Token geschrieben, das vom Authentisierungsserver signiert und vom Reverse Proxy jedem Aufruf mitgegeben wird. Das Secure Token garantiert die Authentizität des Aufrufs, aus diesem Grund wird seine Signatur von allen an der Kommunikation beteiligten Knoten überprüft.

Damit die Kommunikationskette über sämtliche Verbindungen lückenlos abgesichert ist, werden die daran beteiligten Anwendungs- und Middleware-Komponenten über einen so genannten Security Stack in eine einheitliche Sicherheitsschicht eingebunden.

> Der modulare Aufbau erlaubt die flexible Integration verschiedener Authentisierungsmechanismen.

## Portal-spezifische Komponenten

Die grosse Anzahl und Heterogenität der Systeme, die integriert werden mussten, machten die Entwicklung spezieller Komponenten für das SSO-Portal des EJPD notwendig. Bereits vorhandene, Web-basierte Applikationen können ohne Code-Anpassungen in das Portal eingebunden werden. Requests an solche Anwendungen werden vom Reverse Proxy mit allen wichtigen Informationen angereichert, so dass die Applikation aufgerufen werden kann, beispielsweise indem die Benutzerkennung und das Passwort in das entsprechende Header-Feld eingefügt werden. Für die Applikation selbst ist dieser Vorgang vollständig transparent, das heisst die Tatsache, dass sie hinter einem Portal läuft, hat keine Auswirkungen auf die Applikation.

Der gewählte Lösungsansatz erlaubt es, die Sicherheit noch weiter zu erhöhen, indem das jedem Request mitgegebene Security Token in den Applikationen selbst verifiziert wird. Damit die Anwendungen diesen zusätzlichen Schritt ausführen können, muss entweder das dafür vorgesehene Nevis-Modul konfigurativ eingebunden oder der Code der einzelnen Applikationen um den Validator erweitert werden.

## Die Integration von Rich Clients

Für die Integration von Rich Clients, die mit einem proprietären Protokoll arbeiten, kommt auf Seiten des Client ein lokaler Smart Proxy zum Einsatz. Dessen Aufgabe ist es, die Applikation aufzustarten und die sichere Verbindung zum Portal zu unterhalten. Auf Portalseite wird eine Tunneling-Komponente verwendet, die speziell für das SSO-Portal des EJPD entwickelt wurde. Diese Komponente macht es möglich, die proprietären Protokolle zu integrieren, und sorgt für die authentische Koppelung an den Reverse Proxy.

## Tunneling

Erst die Tunneling-Komponente macht aus dem SSO-Portal eine multiprotokollfähige Lösung. Da einzelne der eingebundenen Applikationen Voll-Duplex-Kommunikation verlangen, beispielsweise für asynchrone Notifikation und Alarming, konnte kein gewöhnliches HTTPS-Tunneling eingesetzt werden. Der speziell für dieses Projekt realisierte Tunnel lässt nur authentische Verbindungen zu, das bedeutet, dass die Verbindung mit dem Server einer Legacy-Applikation erst hergestellt werden kann, nachdem sich ein Benutzer erfolgreich im Portal angemeldet hat.

Wird vom Portal aus eine Legacy-Applikation aufgerufen, erkennt der Smart Proxy dies anhand der Angaben im Request und gibt den beim Login aufgebauten SSL-Kontext an den Tunnel weiter (SSL Reuse). Der lokale Smart Proxy startet die gewünschte Applikation und baut die Verbindung zum Tunnel auf. Anhand der SSL Session ID überprüft die Tunnel-Komponente, ob der Benutzer authentisiert ist und stellt die Verbindung zum entsprechenden Zielsystem her.

Im Folgenden verläuft die gesamte Kommunikation zwischen Client und Server über den Smart Proxy. Dabei kann via Konfiguration der Tunnel-Komponente zwischen zwei Arbeitsweisen gewählt werden. Wird vollständig transparent gearbeitet, wird der Request ohne Anreicherung weitergegeben. Soll die Sicherheit noch weiter erhöht werden, kann das Security Token vor der ersten Datenübertragung mitgesendet werden. In diesem Fall muss die Applikation auf dem Host allerdings so angepasst werden, dass sie in der Lage ist, das Security Token auszuwerten.

> Zur erfolgreichen Realisierung des Projektes hat der Einsatz von erweiterbaren Standard-Komponenten wesentlich beigetragen.

Bei der Umsetzung des Single-Signon-Portals für das EJPD hat sich herausgestellt, dass die zentrale Herausforderung in der sicheren Integration ganz unterschiedlicher Protokolle besteht. Zur erfolgreichen Realisierung dieses anspruchsvollen Projekts in nur sechs Monaten hat der Einsatz von erweiterbaren Standard-Komponenten ganz wesentlich beigetragen.

### Andreas Signer

*Andreas Signer ist diplomierter ETH-Software-Ingenieur und arbeitet seit rund drei Jahren in der AdNovum. Als technischer Projektleiter war er für die Realisierung des Single-Signon-Portals für das EJPD verantwortlich. Gegen Ende dieses Jahres wird er seinen Zürcher Arbeitsplatz mit einen amerikanischen vertauschen und die Leitung der AdNovum-Niederlassung in San Mateo übernehmen.*

# Kundenverhalten analysieren

Kundendaten werden heute in rasanter Geschwindigkeit produziert, ausgetauscht und angesammelt. Nun ist eine Software verfügbar, mit der diese bezüglich Geldwäscherei und Wirtschaftskriminalität analysiert werden können.

Bernhard Kunz, CEO kdlabs AG, Zürich

Ob es nun darum geht, bestehenden Kunden das bestmögliche Angebot zu unterbreiten, ihre Bindung an das Unternehmen zu steigern oder ihr Zahlungs-, Kredit- und Betrugsrisiko abzuschätzen: Im Kern geht es immer darum, ob ein Unternehmen seine Kunden kennt, und deren Potenzial und Risiko präzis lokalisieren kann. Nur, wenn es zwischen Unternehmen und Kunden an persönlichen Kontakten mangelt, ist das «Know Your Customer»-Prinzip meist nicht einfach umzusetzen. In diesen Fällen sind analytische Verfahren gefragt, die aus den Kundendaten, bekannten Verhaltensmustern und Ähnlichkeiten mit bereits identifizierten Vorfällen Entscheidungsgrundlagen liefern. Die kdlabs AG hat mit kdprevent™ eine spezielle Software für Banken entwickelt, die bezüglich Geldwäscherei und Wirtschaftskriminalität laufend Entscheidungsgrundlagen liefert.

> Spezielle Analyseverfahren erlauben es, Geschäfte mit kriminellem Bezug zu verhindern.

## Geldwäscherei wirksam begegnen

Um verdächtige Kunden und Transaktionen frühzeitig zu erkennen, Risikogruppen zu überwachen und Geschäfte mit kriminellem Bezug nach Möglichkeit zu verhindern, setzt kdprevent™ verschiedene hochentwickelte Analyseverfahren ein, die bedürfnisgerecht kombiniert und stufenweise ausgebaut werden können:

– Blacklist-Matching: Verfügbar sind Ausbaustufen vom standardmässigen Abgleich mit öffentlichen Namenslisten bis hin zur Einbindung von weiteren Datenquellen und Spezialprodukten über vorbereitete Schnittstellen.

– Anwendung von Regeln: Möglich ist die Definition von einfachen Regeln zur Transaktions- und Kundenüberwachung bis hin zu komplexen Regelwerken.

– Mustererkennung: Zur Aufdeckung von verdächtigen, bisher unbekannten Verhaltensweisen sind induktive Analyseverfahren und Data Mining einsetzbar.

Zusätzlich zur Identifikation von Verdachtsmomenten unterstützt kdprevent™ die Investigation, Weiterverarbeitung und Dokumentation von identifizierten Fällen. Eine beliebige Anzahl von Benutzern sind über einen gemeinsamen, einfach zu bedienenden Workflow miteinander verbunden. Jeder einzelne Arbeitsschritt wird lückenlos und revisionskonform dokumentiert und kann zu jedem Zeitpunkt mit vorbereiteten Reports überprüft und nachvollzogen werden.

Der modulare Aufbau von kdprevent™ erlaubt es, Investitionen auf die Komponenten auszurichten, die ein Finanzdienstleister aktuell benötigt. Gleichzeitig lassen optionale Zusatzkomponenten jederzeit eine Verstärkung der Abwehr gegen Geldwäscherei und Wirtschaftskriminalität zu. Über die Vermeidung von Haftungsklagen, Reputationsschäden und finanziellen Verlusten hinaus stellt kdprevent™ sicher, dass die interne Abwehr gegen Geldwäscherei effizient abläuft, dass unbegründete Alarme minimiert und die Betriebskosten dadurch optimiert werden.

## Datensicherheit gewährleisten

Wesentlich für die Aufdeckung von Finanzkriminalität ist der Zugang zu hochsensiblen Kundendaten wie etwa Namen, die hinter anonymen Kontoverbindungen und Transaktionen verborgen sind. Diese Art von Information, die selbst bankintern oft nicht allen Mitarbeitenden zur Verfügung steht, verlangt gerade bei sehr grossen, dezentral organisierten Finanzinstituten spezielle Vorkehrungen bezüglich Datenzugriff und -sicherheit. Mit Einbezug der Erfahrung der AdNovum werden auch diese Aspekte optimal abgedeckt. ■

### kdlabs AG

*Die kdlabs AG ist in der Schweiz führend in der Analyse von grossen und komplexen Datenbeständen zur Optimierung von Kundenrisiken und zur Ausschöpfung des Kundenpotenzials. Auf der Basis gezielter Anwendungen wie Verkaufs- und Kündigungsprognosen oder Zahlungs- und Kreditrisiken entwickelt die kdlabs AG auch Software-Produkte, z. B. kdprevent™ zur Identifikation von Verdachtsmomenten in den Bereichen Geldwäscherei und Wirtschaftskriminalität.*
*Die 2000 gegründete kdlabs AG mit Sitz in Zürich hat sich mit kdprevent™ in kurzer Zeit als Schweizer Anbieter einer Anti-Geldwäscherei-Lösung etabliert. Um mit kdprevent™ auch höchsten Ansprüchen bezüglich Performance und Sicherheit zu genügen, kooperiert die kdlabs AG mit der AdNovum Informatik AG.*


### Impressum

**Herausgeber:**
AdNovum Informatik AG
Corporate Marketing
Röntgenstrasse 22
CH-8005 Zürich
Telefon 044 272 61 11
Telefax 044 272 63 12
E-Mail info@adnovum.ch
www.adnovum.ch
**Verantwortlich und Redaktion:**
Barbara Stammler, Manuel Ott
**Gestaltung und Realisation:**
Rüegg Werbung, Zürich
**Fotografie:**
Nicolas Monkewitz, Zürich