www.bt-magazin.de 1.2015 Heft 20

€ 4,90

S&S

Business Technology

Business Technology

Lean Innovation & IT Leadership

Qualität und Services

Pünktlich und zuverlässig liefern

Qualitätssicherung von morgen

Microservices: Architektur agil skalieren

©iStockphoto.com/tapilipa

Single Sign-On (SSO) mit SAML implementieren

Anwendungssicherheit mit SAML

Unsere Welt ist seit Jahren durch eine wachsende Flut an Software geprägt. Große Unternehmen nutzen heute insbesondere Webapplikationen, um schneller und flexibler auf die Anforderungen der Märkte zu reagieren. Viele davon sind zu unverzichtbaren Bausteinen der Geschäftsprozesse geworden. Aus diesem Grund sind Interoperabilität und Integration von großer Wichtigkeit. Internet- und Cloud-Architekturen tragen dazu bei, dass man fast von jedem Punkt der Erde Zugriff auf diese Applikationen erhält. Doch damit steigt auch das Risiko von unberechtigtem Zugriff und Missbrauch.

AUTOREN: MAMOON YUNUS UND DR. DIRK KRAFZIG

Die typische Herangehensweise an die nötigen Schutzmaßnahmen ist meist simpel. Jede Anwendung erhält eine Login-Maske, die Benutzername und Kennwort abfragt. Obwohl dies auf den ersten Blick eine logische und einfache Methode zu sein scheint, ist sie in der Praxis mit hohem Aufwand verbunden und kann zu unvermeidbaren Sicherheitslücken führen. Indem man den einzelnen Applikationen aufbürdet, Benutzeridentitäten zu speichern und diese unter Umständen mit anderen Benutzerdatenbanken zu synchronisieren, schafft man einen eklatanten Schwachpunkt. Mit der wachsenden Anzahl an öffentlich zugänglichen Applikationen steigt die Wahrscheinlichkeit für Sicherheitslücken. Gelingt es einem Angreifer, eine dieser Anwendungen zu kompromittieren, so ist die Sicherheit aller Anwendungen gefährdet.

Artikelserie

Teil 1: Anwendungssicherheit mit SAML – Eine Einführung
Teil 2: Implementierung von SAML SSO in der Praxis

FEDERATED IDENTITY MANAGEMENT

Die daraus folgenden Herausforderungen führen geradewegs zu Lösungsansätzen, die sich einer Disziplin bedienen, die unter der Bezeichnung Federated Identity Management (FIdM) unterschiedliche Standards und Vorgehensweisen umfasst. FIdM verwendet standardisierte Policies und Protokolle, um einzelnen Benutzern den Zugriff auf verschiedene Anwendungen zu ermöglichen, wenn diese von unterschiedlichen Systemen bereitgestellt werden. Ein besonderer Vorteil von FIdM ist es, dass Geschäftspartner gegenseitig auf Anwendungen

zugreifen können, ohne dafür Benutzerdatenbanken austauschen zu müssen. Dabei ist das Single Sign-On ein angenehmes Nebenprodukt. Single Sign-On (SSO) ist die Fähigkeit von Benutzern, sich mit einem Satz an Anmeldedaten in mehrere Anwendungen einzuloggen.

Obwohl es zahlreiche Standards wie OAuth, OpenID oder proprietäre Lösungen für FIdM gibt, setzt sich Security Assertion Markup Language (SAML) als die vielseitigste und am meisten genutzte Lösung durch.

Dieser Artikel beschreibt, was SAML ist, zeigt ihre Vorteile und Funktionsweise auf und wie SAML eingesetzt werden kann, um Single Sign-On (SSO) sowie Limitationen von SAML zu implementieren, sowie Limitationen von SAML.

WAS IST SAML?

SAML wurde vom OASIS Security Services Technical Committee (SSTC) definiert. Es ist ein XML-basiertes Framework, das verwendet werden kann, um Benutzer zu authentisieren und autorisieren sowie Attribute und Privilegien von Benutzern zu kommunizieren. Dabei ist es nicht notwendig, dass kritische Daten wie zum Beispiel Kennworte oder Benutzerkennungen zwischen Anwendungen ausgetauscht werden. Der SAML-Standard ist durch mehrere Iterationen gegangen und wird heute als ein stabiles, erprobtes und reifes Framework angesehen. Die erste Version von SAML, V1.0, wurde im November 2002 veröffentlicht. Die aktuelle Version, V2.0, stammt von März 2005. Die letzten freigegebenen Änderungen am Standard (Errata) stammen von Mai 2012.

DIE VORTEILE VON SAML

SAML ist in großen Unternehmen vor allem aus drei Gründen beliebt: Es ist standardisiert, sicher und ermöglicht eine einfache Handhabung durch Benutzer.

Durch die Standardisierung ermöglicht SAML die Interoperabilität verschiedener Anwendungen unabhängig von der eingesetzten Technologie. Es ermöglicht offene Architekturen und Identity Federation, ohne dass die Nachteile von geschlossenen, proprietären und herstellerspezifischen Lösungen in Kauf genommen werden müssen, oder dass ein Integrationsmehraufwand entsteht.

Besonders bei unternehmenskritischen Anwendungen ist Sicherheit das A und O. SAML kann dafür eingesetzt werden, eine zentrale Authentifizierung zu implementieren. Dabei kann eine zentrale Benutzerdatenbank verwendet werden, die in höchstem Maße abgesichert wird. Originäre Benutzerdaten können hinter der Firewall verbleiben, während SAML Benutzeridentitäten verifiziert. Das bedeutet, dass Anwendungen keine Benutzerdaten speichern oder synchronisieren müssen. Das wiederum bedeutet, dass es weniger Orte gibt, auf die sich die Schutzmaßnahmen konzentrieren müssen. SAML bietet seinerseits einen hohen Grad an Schutz durch die Verwendung von Public-Key-Infrastrukturen (PKI).

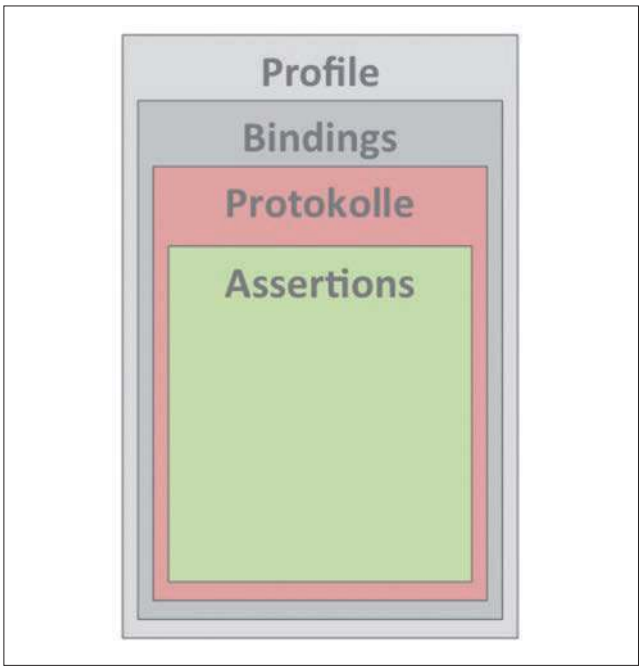

Abb. 1: Konzepte von SAML

Viele Anwender halten die Benutzerfreundlichkeit für eine der größten Stärken von SAML. SAML kann Eigenschaften und Rechte von Benutzern an einen Service Provider (einer Anwendung) sicher weitergeben. Dazu verwendet SAML so genannte Assertions, eines der stärksten Konzepte aus dem SAML-Standard. Assertions ermöglichen es dem Identity Provider (Benutzerdatenbank), detaillierte Informationen über Benutzer bereitzustellen, insbesondere zur Art der Zugriffsrechte, die Benutzer auf verschiedene Anwendungen haben.

WIE FUNKTIONIERT SAML?

Um die Funktionsweise von SAML zu verstehen, müssen drei Parteien eingeführt werden: Benutzer, Identity Provider (IdP) und Service Provider (SP). Der Benutzer repräsentiert die Person, die eine Anwendung verwenden möchte. Der Service Provider repräsentiert die Anwendung, die der Benutzer verwenden will. Der Service Provider wird häufig auch als SAML Relying Party bezeichnet, da der Service Provider auf die Inhalte vertraut, die ihm über SAML bereitgestellt werden. Der Identity Provider speichert die Information über den Benutzer (z. B. Benutzername, Kennwort, x.509, Rollen etc.) und wird häufig auch als SAML Asserting Party bezeichnet.

Das SAML-Framework beherbergt vier wichtige Konzepte: Assertions, Protokolle, Bindings und Profile (**Abb. 1**).

SAML ermöglicht das Anmelden bei mehreren Anwendungen zugleich.

Assertions
Eine SAML-Assertion ist eine XML-Struktur, die Eigenschaften des Benutzers enthält. SAML-Assertions werden durch den Identity Provider erstellt und vom Service Provider konsumiert. SAML-Assertions und die Kontextinformationen erlauben es dem Service Provider, Entscheidungen über Gewährung oder Ablehnung von Zugriffswünschen eines authentifizierten Benutzers zu treffen.

Bevor SAML-Assertions konsumiert werden, überprüft der Service Provider die digitale Signatur (DSIG), um die Integrität und Authentizität des SAML-Tokens zu verifizieren. Sobald der SAML-Token verifiziert ist, analysiert der Service Provider den Inhalt und trifft entsprechende Entscheidungen über Zugriffe. Es gibt drei Typen von SAML-Assertions:

- **Authentication Statements:** Authentication Statements informieren den Service Provider darüber, dass der Benutzer durch den Identity Provider authentifiziert wurde, wann die Authentifizierung stattfand und welche Methode dazu angewandt wurde. Das Authentication Statement kann zusätzliche Information über den Benutzer im Kontext der Authentifizierung enthalten.
- **Attribute Statements:** Mit Attribute Statements liefert der Service Provider Informationen über spezifische, den Benutzer identifizierenden Attribute.
- **Authorization Decision Statements:** Authorization Decision Statements liefern Informationen darüber, welche Ressourcen ein Benutzer verwenden darf.

Hier eine kleine Analogie zur Illustration der Konzepte: Eine Person möchte eine Flasche Schnaps an einem Kiosk kaufen. In dem Beispiel ist die Person, die den Alkohol erwerben möchte, der Benutzer. Der Kiosk ist der Service Provider, der Wohnort der Person, die den Personalausweis herausgegeben hat, der Identity Provider und der Personalausweis entspricht dem SAML-Token. Wie bereits erwähnt, muss der Service Provider die Signatur des SAML-Tokens prüfen, bevor er den Inhalt konsumiert. Das entspricht einer Prüfung, ob der Personalausweis keine Fälschung und nicht abgelaufen ist. Diese Prüfung führt der Kiosk (Service Provider) bzw. der Verkäufer im Kiosk durch scharfes Hinsehen aus. In anderen Anwendungsfällen würde man dazu ausgefeiltere Verfahren anwenden bzw. die Person, die diese Prüfung durchführt, speziell ausbilden. Sobald der Personalausweis als gültig anerkannt ist, wertet der Kiosk die Inhalte (Assertions) des Ausweises aus, um zu entscheiden, ob der Benutzer Alkohol erwerben darf. Dabei wird das Foto herangezogen, um Benutzer und SAML-Token einander zuzuordnen. Zudem wird das Geburtsdatum überprüft, um letztendlich die Autorisierungsentscheidung zu treffen.

Protokolle
Häufig werden die Assertions vom Identiy Provider als Reaktion einer Anfrage durch den Service Provider erstellt. Solche Frage/Antwort-Muster werden SAML-Protokolle genannt. Im Standard gibt es sechs fest definierte SAML-Protokolle, um unterschiedliche Aktionen durchzuführen:

- **Authentication Request Protocol:** Das Authentication Request Protocol wird verwendet, um Assertions über den Benutzer vom Identity Provider zu erhalten. Ziel ist es, einen Security Context mit dem Service Provider zu etablieren. Dieses Protokoll wird im weit verbreiteten Web Browser SSO Profile verwendet.
- **Single Logout Protocol:** Das Single Logout Protocol erlaubt es, alle Sessions eines Benutzers zu terminieren. Das Logout kann durch den Benutzer, den Service Provider oder den Identity Provider eingeleitet werden. Beispielsweise kann das Single Logout Protocol vom Service Provider nach einem Timeout der Benutzersession verwendet werden (Szenario: Benutzer geht in die Mittagspause).
- **Assertion Query and Request Protocol:** Das Assertion Query and Request Protocol definiert Nachrichtentypen und Verarbeitungsregeln, um bereits bekannte Assertions per Referenz abzufragen oder über Suchparameter nach Assertions zu suchen.
- **Artifact Resolution Protocol:** Das Artifact Resolution Protocol ermöglicht es, Referenzen auf SAML-Nachrichten zu versenden, statt die Nachrichten selbst zu übermitteln. Mit diesem Protokoll werden sehr kleine SAML-Artifacts via SAML-Binding anstelle einer größeren SAML-Nachricht verwendet. Das Protokoll stellt

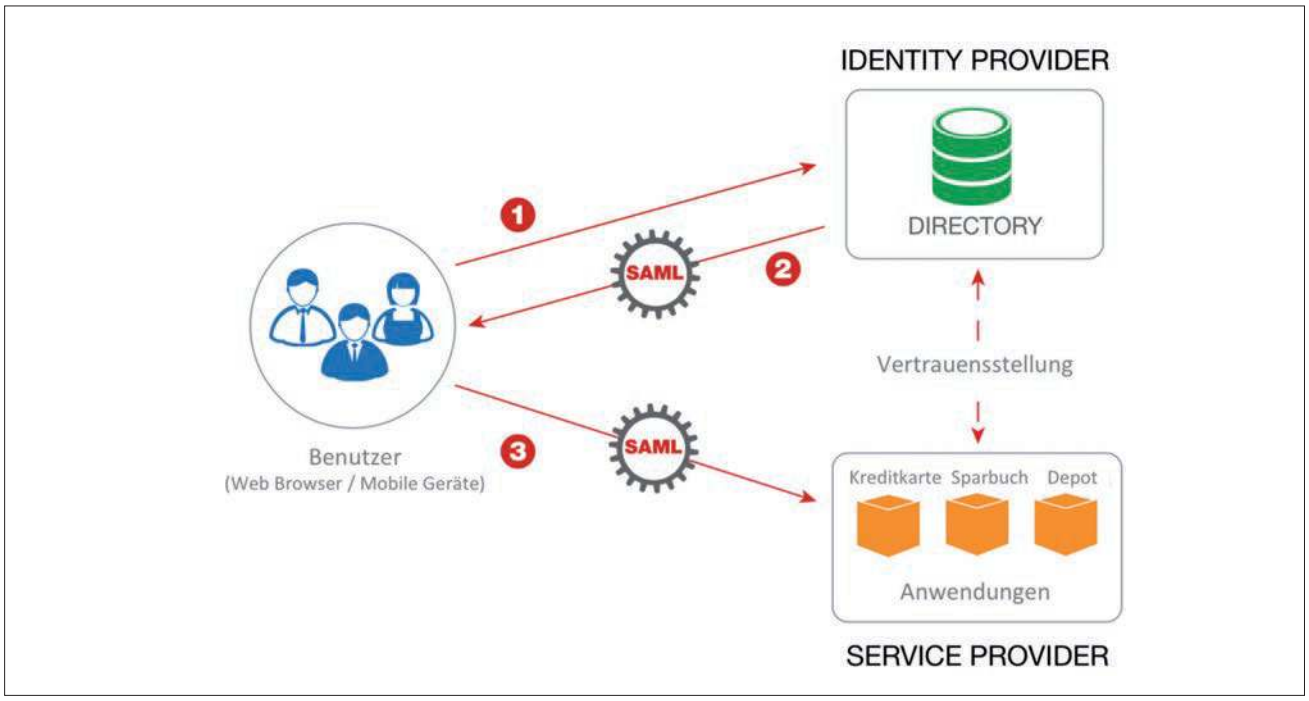

Abb. 2: Schematischer Ablauf von SSO

auch die Möglichkeit bereit, die Referenz aufzulösen und die Originalnachricht zu erhalten. Das Protokoll wird verwendet, wenn die Nachrichtengröße kritisch ist oder wenn man die SAML-Nachrichten über einen separaten sicheren Kanal transportieren möchte.

- **Name Identifier Management Protocol:** Das Name Identifier Management Protocol ermöglicht es, den Wert oder das Format eines Identifiers, der mit einem Benutzer verknüpft ist, zu verändern. Die Anfrage kann vom Service Provider oder vom Identity Provider ausgelöst werden.
- **Name Identifier Mapping Protocol:** Das Name Identifier Mapping Protocol ermöglicht es dem Service Provider in einem Integrationsszenario, einen Identifier für einen Benutzer beim Identity Provider abzufragen, um auf einen weiteren Service Provider zuzugreifen.

Bindings

SAML-Bindings legen fest, wie die Nachrichten im Rahmen der SAML-Protokolle transportiert werden. Typischerweise werden SAML-Protokolle über SOAP oder HTTP transportiert.

Profile

SAML-Profile bündeln Assertions, Protokolle und Bindings für spezifische Anwendungsfälle. Eines der am häufigsten verwendeten SAML-Profile ist das Web Browser SSO Profile.

WIE WIRD SAML FÜR SSP VERWENDET?

SAML ermöglicht es Benutzern, mehrere Anwendungen mit einem einzigen Satz an einmalig eingegebenen Anmeldedaten zu verwenden. Dadurch wird die Benutzung der Anwendungen einfacher und geschäftliche Transaktionen können reibungslos und effizienter durchgeführt werden. Schon heute wird SSO bereits häufig eingesetzt: Sei es, dass wir Onlinebanking machen, mobile Applikationen nutzen oder auf einer Webseite nach komplexen und verschachtelten Informationen suchen.

Wenn sich beispielsweise ein Bankkunde über die Website seiner Bank einloggt, wird er im technischen Sinn regelmäßig unterschiedliche Applikationen verwenden, um auf seine Sparkonten, Kreditkarteninformationen, die Öffnungszeiten seiner Filiale u. v. m. zugreifen zu können. Verschiedene Bankprodukte (wie Sparkonto oder Kreditkarte) werden in der Regel durch unterschiedliche Backend-Systeme unterstützt. Um dem Kunden eine benutzerfreundliche Bedienung der Website zu ermöglichen, müssen die verschiedenen Anwendungen ohne Medienbrüche zusammenarbeiten. Folglich ist ein sicheres Verfahren erforderlich, dass es dem Benutzer nach dem Login ermöglicht, die Daten aus unterschiedlichen Anwendungen bequem aufzurufen. Und genau das ermöglicht SAML.

Betrachten wir die einzelnen Schritte, die es erfordert, um ein SAML-Token zu generieren und es anschließend

dazu zu verwenden, Zugriff auf eine Anwendung zu erhalten. **Abbildung 2** zeigt die grundlegenden Schritte, wie SSO mit SAML funktioniert:

1. Authentifizierung des Benutzers gegen den Identity Provider unter Verwendung von Einzelfaktor oder Mehrfaktorauthentifizierung.
2. Der Identity Provider gibt ein SAML-Token mit den Assertions an den Benutzer heraus. In mobilen Endgeräten und Webbrowsern wird SAML oft mit BASE64 in die HTML-Response eingebettet.
3. Der Browser des Benutzers wird vom Identity Provider zum Service Provider weitergeleitet. Der Browser des Benutzers stellt seine Anfrage an den Service Provider mit dem eingebetteten SAML-Token. Der Service Provider prüft das SAML-Token und seinen Inhalt, um die Zulässigkeit der Anfrage basierend auf der Vertrauensstellung mit dem Identity Provider zu klären. Der Service Provider stellt den Zugriff auf die verschiedenen Onlinebankinganwendungen basierend auf den SAML-Assertions, die im Token enthalten sind, bereit.

In der Praxis gibt es zwei Spielarten, nämlich „Service Provider initiated SAML“ und „Identity Provider initiated SAML“. Diese unterscheiden sich darin, ob der Benutzer den allerersten Request an den Identity Provider oder den Service Provider sendet. Auf dieses und viele weitere Details werden wir in der nächsten Ausgabe mit dem Artikel „Implementierung von SAML SSO in der Praxis“ eingehen.

SAML SSO ermöglicht Benutzern das Verwenden mehrerer Anwendungen ohne Medienbrüche. Zudem kann mit SAML SSO vermieden werden, dass die teilnehmenden Anwendungen größeren Änderungen unterzogen werden müssen, um sich der SAML-Technologie anzupassen.

LIMITATIONEN VON SAML

Obwohl SAML eine der populärsten Methoden für SSO ist, gibt es doch einige Herausforderungen bei der Implementierung von SAML-basierten Lösungen – insbesondere bei der Anbindung von Legacy-Anwendungen:

- **Endpunkte mit SAML ausstatten:** Um SAML-Assertions zu konsumieren, muss die entsprechende Technologie bereitstehen, die es ermöglicht, die XML-Strukturen des SAML-Tokens zu dekodieren, die jeweils relevanten Aspekte zu parsen und anschließend rollenbasiert Entscheidungen über den Zugriff zu treffen. Das kann schwierig sein, wenn diese Aufgaben in die Hand der Entwickler der Legacy-Backends gelegt werden.
- **PKI und CPU:** Die in den SAML-Assertions einhergehende Vertrauensstellung zwischen Identity und Service Provider setzt die Verarbeitung von digitalen Signaturen voraus. Das erfordert neben dem Fachwissen seitens der Entwickler eine entsprechende CPU-Bandbreite und ein funktionierendes Schlüsselmanagement im Rahmen einer PKI.

Legacy-Anwendungen mit SAML auszustatten, kann zeitaufwändig und kostspielig sein. Aus diesem Grund werden häufig API-Gateways eingesetzt, die diese Probleme out of the box lösen. Dadurch wird vermieden, dass bestehende Anwendungen auf Codeebene verändert werden müssen, und trotzdem ein zukunftsweisender Ansatz basierend auf SAML SSO vorangetrieben werden kann.

Links & Literatur

[1] Security Assertion Markup Language (SAML) 2.0 Technical Overview: https://www.oasis-open.org/committees/download.php/11511/sstc-saml-tech-overview-2.0-draft-03.pdf

[2] SAML Executive Overview:https://www.oasis-open.org/committees/download.php/11785/sstc-saml-exec-overview-2.0-draft-06.pdf

[3] Advantages of SAML: http://saml.xml.org/advantages-saml

Mamoon Yunus

ist CEO von Crosscheck Networks, einem führenden Technologieanbieter für Cloud- und Web-Services-Infrastrukturen. Als SOA Pionier und Gründer von Forum Systems hat er wichtige Techniken für XML Appliances patentieren lassen. Er besitzt zwei Abschlüsse vom MIT. InfoWorld hat ihn 2004 als einen von vier „Up and coming CTOs to watch“ ausgezeichnet.

Dr. Dirk Krafzig

ist Gründer von SOAPARK. Als Sprecher auf Konferenzen und Autor von Artikeln und Büchern gilt Dr. Krafzig als ein Protagonist der serviceorientierten Architektur (SOA) und hat maßgeblich zu der Begriffsbildung in diesem Bereich beigetragen. Insbesondere die SOA-Fallstudien mit der Deutschen Post, Credit Suisse, Halifax Bank of Scotland und Winterthur Versicherung in seinem Bestseller „Enterprise SOA“ haben viel Aufmerksamkeit auf sich gezogen. Derzeit arbeitet Dr. Krafzig in einem strategischen SOA-Programm bei einem Mobilfunkanbieter an dem Thema Security.