Föderiertes Identitätsmanagement in der Cloud

Bernd Zwattendorfer[1] · Klaus Stranacher[1] · Felix Hörandner[2]

[1]E-Government Innovationszentrum (EGIZ)
{Bernd.Zwattendorfer | Klaus.Stranacher}@egiz.gv.at

[2]Technische Universität Graz
felix.hoerandner@student.tugraz.at

Zusammenfassung

Ein sicheres Identitätsmanagement sowie eine sichere Authentifizierung sind wesentliche Bestandteile um den Zugriff auf Online Applikationen zu schützen bzw. zu regeln. Nachdem Cloud Computing immer mehr an Bedeutung gewinnt, ist ein sicheres Identitätsmanagement auch im Cloud Kontext essentiell. Im Laufe der Jahre sind bereits einige Identitätsmanagement-Modelle für die Cloud entstanden. Die meisten dieser Modelle sind aber sehr eingeschränkt bei der Wahl des Cloud Service Providers und vernachlässigen Datenschutzaspekte für Benutzerinnen bzw. Benutzer. Aus diesem Grund wird in diesem Beitrag ein neues Identitätsmanagement-Modell für die Cloud vorgestellt, welches Flexibilität bei der Wahl des Cloud Service Providers für Benutzerinnen bzw. Benutzer bietet und sensible Identitätsdaten in der Cloud ausschließlich verschlüsselt verarbeitet. Um diese Eigenschaften umzusetzen, besteht das Modell aus einer Föderation von mehreren Cloud Identity Brokern. Um die Praktikabilität des Modells zu demonstrieren, wurde es prototypisch unter Verwendung standardisierter Protokolle implementiert.

1 Einleitung

Identitätsmanagement umfasst im Wesentlichen die sichere Verwaltung von Identitäten bzw. Benutzerkonten, die Verwaltung zugehöriger Attribute, sowie die sichere Identifizierung und Authentifizierung von Benutzerinnen und Benutzern an Applikationen [BeTa11]. Identitätsmanagement und insbesondere ein sicheres Identitätsmanagement ist ein wesentlicher Kernaspekt bei der Entwicklung bzw. beim Schutz von Zugriffen auf eine Online Applikation.

Aus diesem Grund existieren unterschiedliche Ansätze und Modelle bereits seit einigen Jahren. Im Unternehmensbereich sind Ansätze auf Basis von LDAP (Lightweight Directory Access Protocol) [Scib06] Verzeichnissen oder Kerberos [NYHR05] immer noch präsent. Im Web, und vor allem für ein Identitätsmanagement über Domänengrenzen hinweg, haben sich über die letzten Jahre vor allem die Standards SAML (Security Assertion Markup Language) [LCR+08], OpenID [OpenID], sowie OAuth [Hard12] (im Zusammenhang mit OpenID Connect [SBJ+14]) durchgesetzt.

Die meisten Identitätsmanagementkonzepte verfolgen dabei alle einen ähnlichen architektonischen Ansatz, wobei ein *Identity Provider*, ein *Service Provider*, und eine *Benutzerin* bzw. ein

P. Schartner · P. Lipp (Hrsg.) · D•A•CH Security 2014 · syssec (2014) 248-259.

Benutzer als Stakeholder involviert sind [BeTa11]. Der Service Provider stellt dabei schützenswerte Daten oder Ressourcen über eine Applikation zur Verfügung, auf die eine Benutzerin bzw. ein Benutzer zugreifen möchte. Um den Aufwand für einen Zugriffsschutz für einen Service Provider möglichst gering zu halten, wird diese Funktion an den Identity Provider ausgelagert. Der Identity Provider verwaltet dabei die Benutzerkonten bzw. die digitale Identität der Benutzerin bzw. des Benutzers. Zusätzlich regelt der Identity Provider auch die Identifizierung und Authentifizierung von Benutzerinnen und Benutzern. Der Identity Provider übernimmt also alle Funktionen zur Regelung des Zugriffsschutzes für den Service Provider. Der Service Provider erhält im Rahmen eines Identifizierungs- und Authentifizierungsprozess nur mehr die Identitäts- und Authentifizierungsdaten einer Benutzerin bzw. eines Benutzers vom Identity Provider. Basierend auf diesen Daten kann der Service Provider dann den Zugriff auf die geschützte Ressource gewähren oder verweigern. Abbildung 1 zeigt diese typische Identitätsmanagement-Architektur.

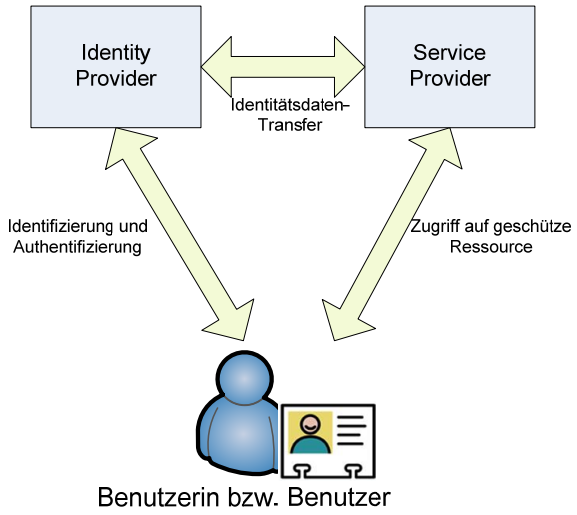

Abb. 1: Identitätsmanagement-Architektur

Aufgrund der immer und stetig steigenden Anzahl von Cloud Applikationen ist ein sicheres Identitätsmanagement auch ein essentielles Thema im Rahmen von Cloud Computing. Existierende Identitätsmanagement-Modelle sind jedoch nicht ad-hoc im Cloud Computing anwendbar, da im Rahmen von Cloud Computing zusätzliche Anforderungen (z.B. Datenschutz) berücksichtigt werden müssen. Einige Identitätsmanagement-Modelle sind jedoch bereits für die Cloud und deren Eigenschaften und Anforderungen entwickelt worden. Ein paar dieser Modelle werden in Abschnitt 2 genauer vorgestellt.

Wesentlicher Beitrag dieser Arbeit ist die Vorstellung eines neuen Identitätsmanagement-Modell für die Cloud, welches einerseits die Vorteile des Cloud Computing ausnützt, und anderseits dessen Schwachstellen entsprechend berücksichtigt. Dieses Modell und eine prototypische Implementierung werden in den Abschnitten 3 und 4 genauer beschrieben. Das beschrieben Modell wird abschließend entsprechend diskutiert.

2 Identitätsmanagement-Modelle in der Cloud

Identitätsmanagement und im Speziellen ein sicheres Identitätsmanagement spielen auch im Cloud Computing eine wichtige Rolle. Die Idee der Verwendung von Identitätsmanagement-

Konzepten auch in der Cloud ist nicht neu, da viele Unternehmen und Organisationen ihre Applikationen in die Cloud migrieren, um beispielsweise die Kostenvorteile oder die höhere Skalierbarkeit und Dynamik einer Cloud voll ausschöpfen zu können. [Gopa09, Cox12, Goul10, ZwZS14] haben bereits unterschiedliche Identitätsmanagement-Modelle in der Cloud klassifiziert. Basierend auf deren Arbeiten werden im Folgenden einige dieser Modelle vorgestellt.

2.1 Identitätsmanagement in der Cloud

Das „*Identitätsmanagement in der Cloud*"-Modell ist das einfachste Modell. In diesem Modell betreibt ein Cloud Service Provider, z.B. Google, sowohl den Identity Provider als auch die Applikation, die die schützenswerten Daten bereithält (Service Provider). Im Prinzip verschmelzen Identity Provider und Service Provider in diesem Modell in der Cloud. Abbildung 2 illustriert dieses Modell.

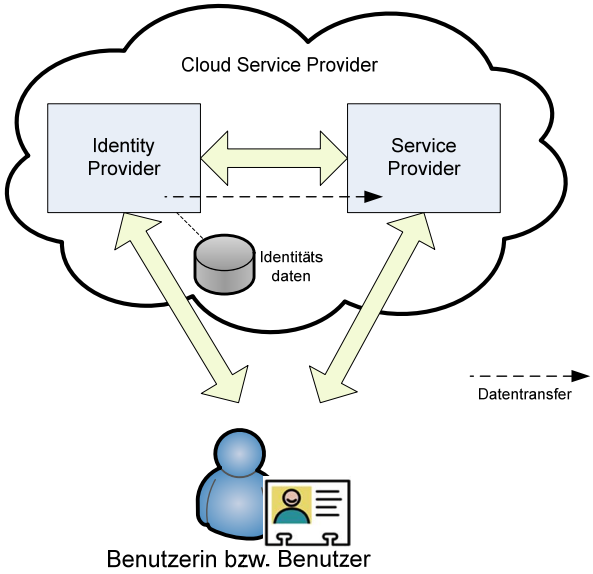

Abb. 2: Identitätsmanagement in der Cloud -Modell

Der Vorteil dieses Modells ist, dass Unternehmen bereits ein fix fertiges Identitätsmanagement, nämlich das vom Cloud Service Provider für seine Applikationen bereitgestellte, verwenden können. Um die Speicherung und das Management der Identitätsdaten kümmert sich der Cloud Service Provider. Nachteil ist natürlich, dass nur jene Daten verwendet werden können, die der Cloud Service Provider auch unterstützt. Außerdem liegt die Kontrolle des Identitätsmanagementsystems (Identity Provider) komplett beim Cloud Service Provider. Eine Migration oder Synchronisierung externer Identitätsdaten zum Cloud Service Provider ist in diesem Modell nicht vorgesehen. D.h., Benutzerinnen und Benutzer müssen sich beim Cloud Service Provider neu registrieren, wenn sie entsprechende Cloud Services nutzen möchten.

2.2 Identitätsmanagement zur Cloud

In diesem Modell (siehe Abbildung 3) ist die Verwendung eines externen Identitätsmanagementsystems möglich, d.h. der Cloud Provider empfängt nur Identitätsdaten von einem externen Identity Provider. Der Unterschied zum vorherigen Modell ist, dass der Identity Provider nicht in der Cloud betrieben wird, sondern in einem klassischen Rechenzentrum unter vollständiger Kontrolle jener Organisation, die auch die Identitätsdaten bereitstellt und verwaltet. Der

Cloud Provider hostet in diesem Fall nur die Applikation, an der eine Authentifizierung notwendig ist. Der Datentransfer zwischen dem externen Identity Provider und der Cloud Applikation erfolgt üblicherweise über standardisierte Schnittstellen und Identitätsprotokolle, wie z.B. SAML oder OAuth. Google und Salesforce.com sind zwei Vertreter, die dieses Modell unterstützen.

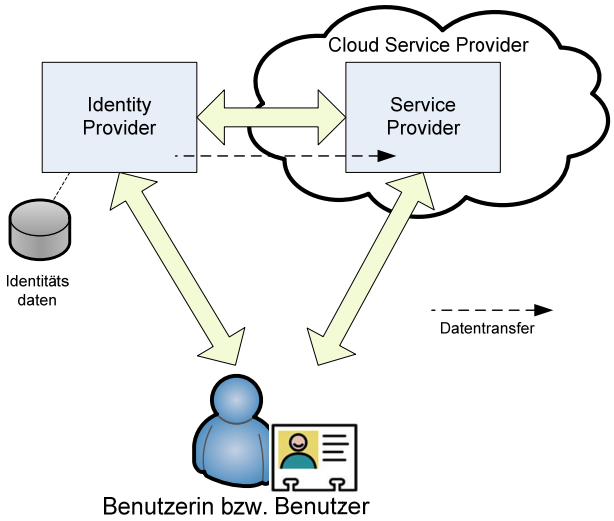

Abb. 3: Identitätsmanagement zur Cloud -Modell

Der Vorteil dieses Modells ist, dass die Kontrolle über die Identitätsdaten nicht zum Cloud Provider transferiert werden muss und die eigene Organisation die Kontrolle behalten kann. Die Identifizierung und Authentifizierung erfolgt beim externen Identity Provider, Identitäts- und Authentifizierungsdaten werden nur über eine entsprechende Schnittstelle an den Cloud Service Provider und dessen Applikation übertragen. Diese Schnittstelle stellt auch den größten Nachteil dieses Modells dar, da Interoperabilitätsprobleme bei der Implementierung sowohl auf Seiten des Cloud Service Providers als auch Seiten des externen Identity Providers auftreten können. Außerdem werden in diesem Modell nicht die vollen positiven Cloud Eigenschaften ausgenützt, da der Identity Provider nicht in der Cloud betrieben wird.

2.3 Identitätsmanagement von der Cloud

In diesem Modell werden sowohl der Identity Provider als auch der Service Provider in der Cloud betrieben. Wesentlicher Unterschied zum *„Identitätsmanagement in der Cloud“*-Modell ist, dass der Identity Provider und der Service Provider von zwei unterschiedlichen Cloud Service Providern betrieben werden. D.h., ein Cloud Service Provider hostet den Identity Provider und ein anderer Cloud Service Provider die Applikation. Diese Trennung spiegelt auch die generelle Architektur aus Abbildung 1 wider, nur dass beide Provider (Identity Provider und Service Provider) in der Cloud betrieben werden, was die Vorteile des Cloud Computing effizient ausnützt. Abbildung 3 zeigt dieses Modell.

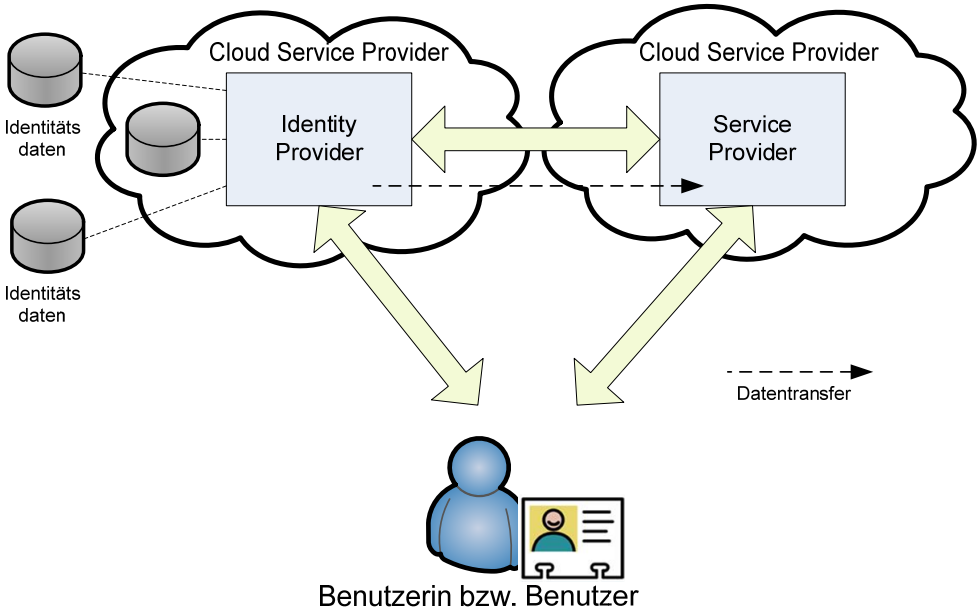

Abb. 4: Identitätsmanagement von der Cloud -Modell

Neben den klassischen Vorteilen des Cloud Computing ist der wesentliche Vorteil dieses Modells, dass sich eine Organisation den Identity Provider in der Cloud, dem sie vertrauen möchte, aussuchen kann. Dies ist von besonderer Wichtigkeit, da die Identitätsdaten der Organisation vollkommen vom Cloud Service Provider verwaltet werden, und nicht wie bei beim *„Identitätsmanagement zur Cloud"*-Modell in einem externen, von der Organisation kontrollierbaren Rechenzentrum. Dies ist allerdings auch zugleich der größte Nachteil, da Identitätsdaten und somit sensible Daten in der Cloud gespeichert werden. Insbesondere der Datenschutz ist hier gefährdet, da Datenschutz eines der Hauptprobleme im Bezug zum Cloud Computing, im Besonderen in der Public Cloud, darstellt [PeBe10].

2.4 Cloud Identity Broker-Modell

Dieses Modell stellt im Prinzip eine Erweiterung des *„Identitätsmanagement von der Cloud"*-Modell dar. Im Gegensatz zu einem einzelnen Identity Provider wird ein sogenannter Identity Broker in der Cloud betrieben. Ein Identity Broker ist eine Art Hub zwischen ein oder mehreren Service Providern und ein oder mehreren Identity Providern. In anderen Worten, über einen Identity Broker kann ein Service Provider unterschiedliche Identity Provider unterstützen und eine Benutzerin bzw. ein Benutzer kann sich bei einer Authentifizierung den gewünschten Identity Provider aussuchen. Ein Projekt, welches auf diesen *„Cloud Identity Broker"*-Ansatz basiert, ist SkIDentity [HSW+12]. Abbildung 5 veranschaulicht das Cloud Identity Broker-Modell.

Der Vorteil dieses Modells ist, dass die Komplexität einzelner Identity Provider vom Identity Broker gegenüber dem Service Provider „versteckt" wird. Ein Service Provider, der mehrere Identity Provider für eine Benutzerinnen- bzw. Benutzer-Authentifizierung unterstützen möchte, braucht in diesem Modell jedoch nicht alle Schnittstellen zu den einzelnen Identity Providern implementieren, sondern es reicht die Implementierung der Schnittstelle zum Identity Broker. Ein Nachteil dieses Modells ist jedoch, dass sowohl Benutzerinnen bzw. Benutzer als auch der Service Provider ein und demselben Identity Broker in der Cloud vertrauen müssen, um unterschiedliche Identity Provider nutzen zu können. Das heißt auch, dass Benutzerinnen

bzw. Benutzer und Service Provider von der offerierten Funktionalität des Identity Brokers abhängig sind. Möchte beispielsweise eine Benutzerin bzw. ein Benutzer sich bei einem Identity Provider authentifizieren, welcher vom Identity Broker nicht unterstützt wird, so ist eine Anmeldung schlicht und einfach nicht möglich.

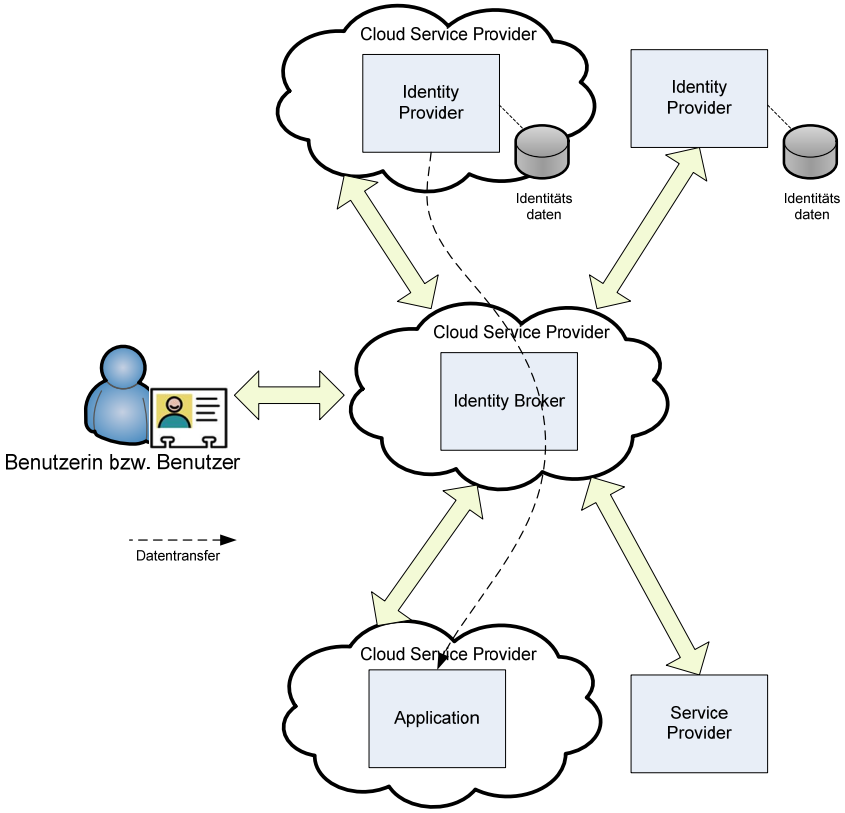

Abb. 5: Cloud Identity Broker-Modell

3 Föderiertes Identitätsmanagement in der Cloud

Um das Problem der Abhängigkeit von ein und demselben Identity Broker sowohl für Benutzerinnen bzw. Benutzer als auch für Service Provider zu lösen, wird in diesem Abschnitt ein neues Identitätsmanagement-Modell für die Cloud basierend auf föderierten Cloud Identity Brokern vorgestellt. In diesem Modell müssen Benutzerinnen bzw. Benutzer und Service Provider nicht auf denselben Cloud Identity Broker für eine Authentifizierung setzen. Beide können ein Vertrags- und Vertrauensverhältnis mit unterschiedlichen Cloud Identity Brokern eingehen, was die Flexibilität erhöht. Die unterschiedlichen Cloud Identity Broker können auch jeweils auf die unterschiedlichen individuellen Bedürfnisse von Benutzerinnen bzw. Benutzern oder von Service Providern eingehen. Beispiele für solche Bedürfnisse wären nationale Regulierungen oder Gesetze. Obwohl kein direktes Vertrauensverhältnis mit ein und demselben Cloud Identity Broker für Benutzerinnen bzw. Benutzer und dem Service Provider wie im Abschnitt 2.4 vorgestellten Modell besteht, so können sich durch Föderation der Cloud Identity Broker Benutzerinnen bzw. Benutzer trotzdem am Service Provider anmelden. Abbildung 5 zeigt dieses föderierte Cloud Identity Broker-Modell.

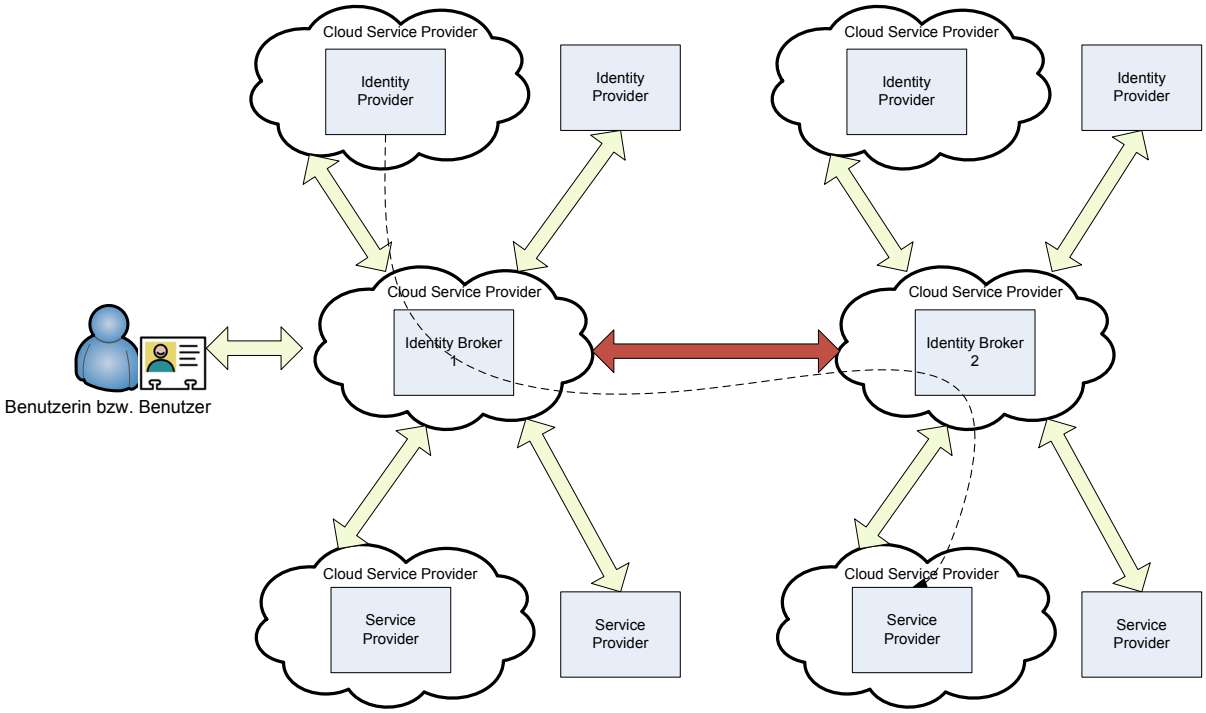

Abb. 6: Föderiertes Cloud Identity Broker-Modell

In diesem föderierten Ansatz ist es möglich, dass der Service Provider eine Vertragsbeziehung mit dem Identity Broker 2 besitzt, während die Benutzerin bzw. der Benutzer hingegen eine Vertragsbeziehung mit dem Identity Broker 1 hat. Beide Identity Broker haben ebenfalls ein entsprechendes Vertrauensverhältnis bzw. eine Vertragsbeziehung untereinander.

Betrachtet man den Informations- und Prozessfluss aus Abbildung 5 etwas genauer, so kontaktiert eine Benutzerin bzw. ein Benutzer in einem ersten Schritt jenen Service Provider, von dem sie bzw. er eine geschützte Ressource oder Service nutzen möchte. Für den Zugriff auf die geschützte Ressource wird von der Benutzerin bzw. dem Benutzer eine entsprechende Authentifizierung benötigt. Die Identifizierung und Authentifizierung wird wie in den meisten anderen Cloud Identitätsmanagement-Modellen an einen Identity Provider, in diesem Fall zuerst an einen Cloud Identity Broker, ausgelagert. In diesem Beispiel besitzt der Service Provider ein Vertragsverhältnis mit dem Identity Broker 2, an den die Identifizierung und Authentifizierung von Benutzerinnen und Benutzern ausgelagert wird. Im Gegensatz zu dem im Abschnitt 2.4 beschriebenen *„Cloud Identity Broker“*-Modell, hat die Benutzerin bzw. der Benutzer in diesem föderierten Modell kein Vertragsverhältnis mit demselben Identity Broker wie die Service Provider (Identity Broker 2), sondern mit Identity Broker 1. Dieser unterstützt im Gegensatz zum Identity Broker 1 jenen Identity Provider, den die Benutzerin bzw. der Benutzer auch für eine Authentifizierung bei dem ausgewählten Service Provider nutzen möchte. Um diesen Identity Provider auch nutzen zu können, wird die Benutzerin bzw. der Benutzer an Identity Broker 1 weitergeleitet. Danach initiiert der Identity Broker 1 den Identifizierungs- und Authentifizierungsprozess mit dem gewünschten Identity Provider. Die Benutzerin bzw. der Benutzer authentifiziert sich dabei beim Identity Provider mit ihrem bzw. seinem gewünschten Authentifizierungsmechanismus. War der Authentifizierungsvorgang erfolgreich, so werden entsprechende Identitäts- und Authentifizierungsdaten der Benutzerin bzw. des Benutzers an den Identity Broker 1 übermittelt. Anschließend leitet Identity Broker 1 die Daten an den Identity Broker 2 weiter, welcher sie schlussendlich an den Service Provider transferiert. Basierend auf den

empfangenden Daten erlaubt bzw. verbietet der Service Provider den Zugriff auf die geschützte Ressource. Im Prinzip gibt es in diesem Modell drei Kommunikationskanäle, wo Identitätsdaten ausgetauscht werden, nämlich zwischen:

1. Identity Provider und Identity Broker 1
2. Identity Broker 1 und Identity Broker 2
3. Identity Broker 2 und Service Provider

Für den Kommunikationskanal zwischen Identity Provider und Identity Broker 1 bzw. Identity Broker 2 und Service Provider können einfach existierende Identitätsprotokolle wie z.B. SAML oder OAuth verwendet werden.

Obwohl das bisher beschriebene Modell gegenüber den anderen Modellen prinzipiell Datenschutz-freundlicher ist[1], besteht trotzdem die Gefahr, dass ein Cloud Provider sensible Identitätsdaten mitlesen und somit ein Benutzerinnenprofil bzw. ein Benutzerprofil erstellen kann. Solange die Daten – auch wenn sie verschlüsselt zum Cloud Service Provider übertragen werden – im Klartext beim Cloud Service Provider gespeichert werden, besteht diese Gefahr. Nachdem Cloud Service Provider ihre gespeicherten Daten üblicherweise auf verschiedene Serverfarmen und auch Länder verteilen können, besteht in so einem Fall auch kein entsprechender Rechtsschutz, wenn Daten bei einem Provider außerhalb der EU abgelegt werden. So sagt beispielsweise der US Patriot Act [US-PA01], dass beliebige Daten, sofern sie von einem US-stämmigen Unternehmen gespeichert werden, von Behörden aus den USA auf Verlangen inspiziert und eingesehen werden können.

Um keine sensiblen Daten im Klartext bei solch einer ungewollten Inspizierung preiszugeben und um den Datenschutz für Benutzerinnen bzw. Benutzer wahren zu können, werden in einer Erweiterung dieses föderierten Cloud Identity Broker-Modell die Daten nur mehr verschlüsselt transferiert bzw. gespeichert. Als Verschlüsselungsalgorithmus wird dabei Proxy Re- Encryption [AFGH06] eingesetzt.

Proxy Re-Encryption ermöglicht es einem Proxy Daten, welche für eine Partei A und dessen Public $Key_{(A)}$ verschlüsselt sind, unter zu Hilfenahme eines so-genannten Re-Encryption $Keys_{(A\rightarrow B)}$ die Daten so umzuschlüsseln, dass sie anschließend von einer Partei B und dessen Private $Key_{(B)}$ entschlüsselt werden können. Der Proxy erhält dabei weder Zugriff die Private Keys von A und B noch auf den Klartext der verschlüsselten Daten. D.h., die Umschlüsselung erfolgt direkt auf den verschlüsselten Daten, eine Entschlüsselung der Daten am Proxy ist nicht notwendig. Für die Erstellung des Re-Encryption $Keys_{(A\rightarrow B)}$ wird der Private Key von A und der Public Key von B benötigt.

Im folgenden Abschnitt wird eine prototypische Implementierung dieses Modelles unter Verwendung von Proxy Re-Encryption beschrieben.

4 Implementierung

Die prototypische Implementierung wurde so umgesetzt, dass ein Identifizierungs- und Authentifizierungsprozess über das vorgestellte Modell verschlüsselt möglich ist. Dabei wurden die folgenden Komponenten umgesetzt (Identity Provider, Identity Broker 1 und 2, Service Provider, Re-Encryption Key Generator), welche auch kurz in Abschnitt 4.1 beschrieben werden.

[1] Identity Broker 1 erfährt beispielsweise nie, bei welchem Service Provider eine Benutzerin bzw. ein Benutzer sich genau anmelden will. Identity Broker 2 agiert hier immer als Intermediär dazwischen.

Abbildung 7 zeigt die Komponenten der prototypischen Umsetzung. Die Ziffern beschreiben den Prozessfluss, welcher in Abschnitt 4.2 beschrieben wird.

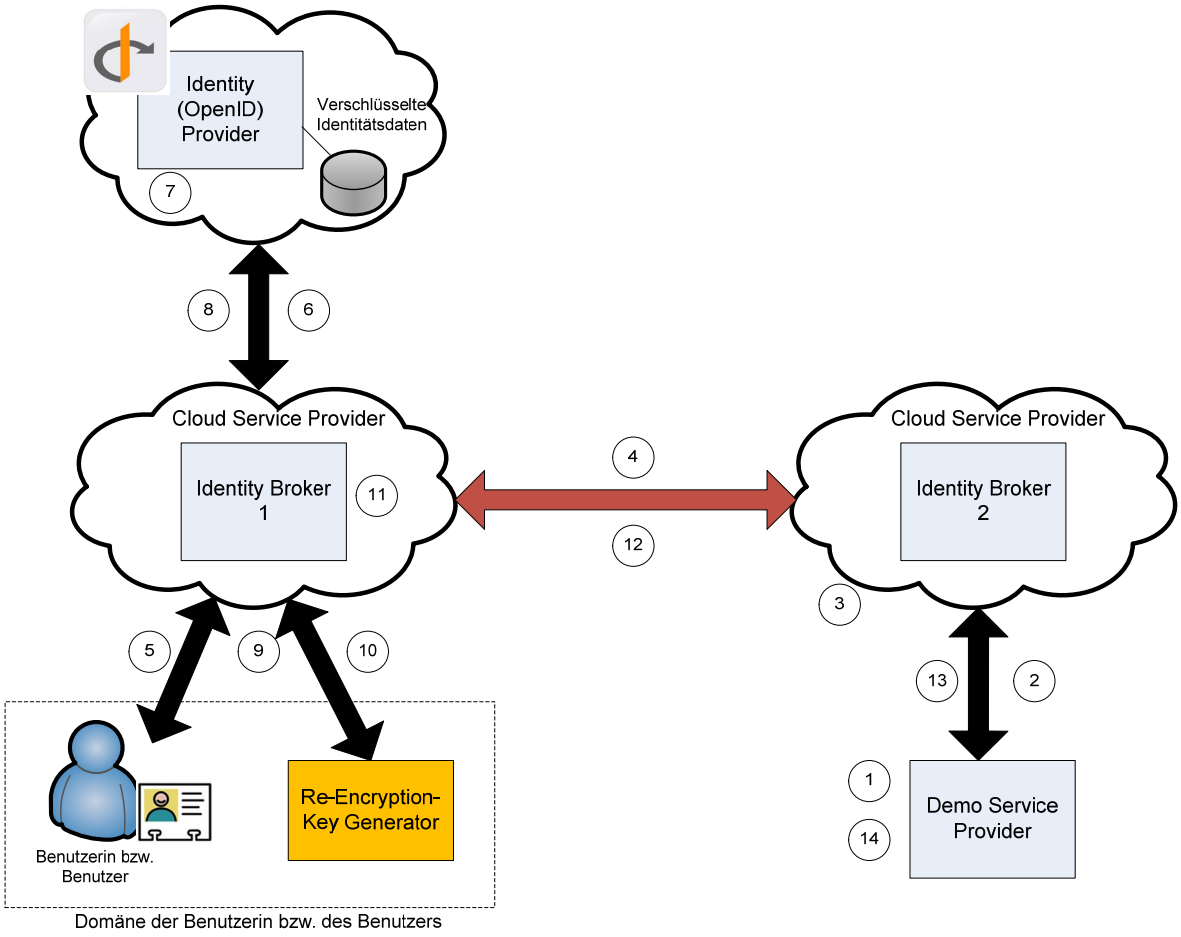

Abb. 7: Prototypische Implementierung

4.1 Komponenten

Im Folgenden werden die einzelnen Komponenten der prototypischen Implementierung kurz beschrieben:

- *Identity Provider*
  Als Identity Provider wurde ein selbstkonfigurierter OpenID Provider verwendet, bei dem die Identitätsdaten für die Benutzerin bzw. den Benutzer verschlüsselt abgelegt werden können. Der Identity Provider kann von Identity Broker 1 über das OpenID-Protokoll angesprochen werden.
- *Identity Broker 1*
  Der Identity Broker 1 hat ein Vertrauensverhältnis mit dem Identity Provider sowie mit dem Identity Broker 2. Der Identity Broker 1 kann für die Benutzerin bzw. für den Benutzer verschlüsselte Daten für den Service Provider umschlüsseln.
- *Identity Broker 2*
  Der Identity Broker 2 hat ein Vertrauensverhältnis mit dem Identity Broker 1 und dem

Service Provider. Wesentliche Funktion ist das Auffinden von Identity Broker 1 und der Datentransfer nach einer Authentifizierung zum Service Provider.

- *Service Provider*
  Der implementierte Service Provider ist im Wesentlichen eine Demo-Applikation, die eine Authentifizierung über den Identity Provider benötigt.
- *Re-Encryption Key Generator*
  Der Re-Encryption Key Generator ist keine Server-Komponente, sondern läuft in der Domäne der Benutzerin bzw. des Benutzers. Mit Hilfe des Private Keys der Benutzerin bzw. des Benutzers und des Public Keys des Service Providers kann ein entsprechender Re-Encryption Key$_{\text{(Benutzerin bzw. Benutzer→Service Provider)}}$ erstellt werden.

Die Vertrauensverhältnisse zwischen den einzelnen Komponenten wurden mittels Zertifikaten einer Public-Key-Infrastruktur (PKI) abgebildet. Für die Kommunikation zwischen den einzelnen Komponenten wurden unterschiedliche Protokolle eingesetzt. Alle Kommunikationskanäle wurden zusätzlich mit dem sicheren Transportprotokoll SSL/TLS abgesichert. Die folgenden Protokolle wurden dabei implementiert:

- Identity Provider ↔ Identity Broker 1: OpenID 2.0
- Identity Broker 1 ↔ Identity Broker 2: SAML 2.0
- Identity Broker 2 ↔ Service Provider: SAML 2.0
- Identity Broker 1 ↔ Re-Encryption Key Generator: Eigenes Web Service-Protokoll

4.2 Prozessfluss

Im Folgenden werden die einzelnen Prozessschritte eines Authentifizierungsprozesses anhand der prototypischen Implementierung beschrieben. Für die Beschreibung des Authentifizierungsprozesses müssen zuvor die folgenden Annahmen getroffen werden:

- Die Identitätsdaten sind beim OpenID Provider (Identity Provider) bei der Benutzerinnen- bzw. Benutzerregistrierung für die Benutzerin bzw. den Benutzer verschlüsselt abgelegt worden.
- Bilaterale Vertrauensverhältnisse existieren zwischen
  - Identity Provider und Identity Broker 1
  - Identity Broker 1 und Identity Broker 2
  - Identity Broker 2 und Service Provider
  - Identity Broker 1 und Re-Encryption Key Generator
- Die einzelnen Komponenten besitzen entsprechendes Schlüsselmaterial zum Absichern der Kommunikation (Verschlüsselungs- und Signaturschlüsselpaare) sowie zum Abbilden der Vertrauensverhältnisse.

Ein Authentifizierungsprozess läuft nun wie folgt ab:

1. Eine Benutzerin bzw. ein Benutzer möchte auf eine vom Service Provider geschützte Ressource zugreifen.
2. Die Benutzerin bzw. der Benutzer wird zur Authentifizierung an den Identity Broker 2 weitergeleitet.
3. Identity Broker 2 überprüft die Authentifizierungsanfrage vom Service Provider und frägt die Benutzerin bzw. den Benutzer um die Bekanntgabe ihres bzw. seines gewünschten

Identity Brokers 1. Dies erfolgt durch die Eingabe einer URL, z.B. `https://benutzer.identity-broker1.com`.

4. Identity Broker 2 leitet die Authentifizierungsanfrage an den Identity Broker 1 weiter.
5. Identity Broker 1 verifiziert die Anfrage und bittet die Benutzerin bzw. den Benutzer zur Bekanntgabe ihres bzw. seins OpenID-Identifikators.
6. Anhand des OpenID-Identifikators wird die Benutzerin bzw. der Benutzer zum Identity (OpenID) Provider weitergeleitet.
7. Die Benutzerin bzw. der Benutzer authentifiziert sich entsprechend beim OpenID Provider.
8. Die verschlüsselten Identitätsdaten der Benutzerin bzw. des Benutzers werden an den Identity Broker 1 übertragen.
9. Der Identity Broker 1 schickt eine Anfrage zum lokalen Re-Encryption Key Generator zur Generierung eines Re-Encryption Keys$_{\text{(Benutzerin bzw. Benutzer}\rightarrow\text{Service Provider)}}$. Die Anfrage enthält dabei den Public Key des Service Providers, der zur Re-Encryption Generierung notwendige Private Key der Benutzerin bzw. des Benutzers wird lokal von der Benutzerin bzw. dem Benutzer bereitgestellt.
10. Der generierte Re-Encryption Key wird an den Identity Broker 1 retourniert.
11. Der Identity Broker 1 schlüsselt die Identitätsdaten für den Service Provider um.
12. Identity Broker 1 überträgt die umgeschlüsselten Identitätsdaten zu Identity Broker 2.
13. Identity Broker 2 überträgt die umgeschlüsselten Daten zum Service Provider.
14. Der Service Provider entschlüsselt die Daten und basierend auf deren Inhalt wird Zugriff auf die geschützte Ressource gewährt.

5 Zusammenfassung und Diskussion

Im Rahmen dieses Beitrages wurden zuerst unterschiedliche Identitätsmanagement-Modelle für die Cloud beschrieben und deren Vor-und Nachteile diskutiert. Anschließend wurde ein neues Identitätsmanagement für die Cloud vorgestellt, welches mehrere Cloud Identity Broker in einer Föderation zusammenschließt. Dieses Modell wurde auch anhand unterschiedlicher Komponenten prototypisch implementiert. Das vorgestellte Modell bietet Benutzerinnen bzw. Benutzern eine größere Flexibilität hinsichtlich der Auswahl eines gewünschten Cloud Identity Brokers. Die Abhängigkeit von nur einem zentralen Identity Broker wie im klassischen *„Cloud Identity Broker“*-Modell ist nicht mehr gegeben. Zusätzlich bietet dieses Modell entsprechenden Datenschutz in Hinblick auf den Cloud Service Provider, da alle Daten nur verschlüsselt zwischen den einzelnen Komponenten ausgetauscht werden. Die Benutzerin bzw. der Benutzer besitzt dabei immer die volle Kontrolle über ihre bzw. seine Daten, da diese anfänglich nur für sie bzw. ihn verschlüsselt beim Identity Provider abgelegt wurden. Die Erstellung eines Re-Encryption Keys und somit die Umschlüsselung für einen Service Provider erfolgt ebenfalls nur unter der alleinigen Kontrolle der Benutzerin bzw. des Benutzers. Letztendlich ist durch die Verwendung von standardisierten Protokollen wie OpenID oder SAML eine einfache Integration in bestehende Infrastrukturen möglich.

Zukünftige Erweiterungen dieses Modells bzw. der prototypischen Implementierung betreffen die Integration von weiteren Identity Providern, wie z.B. Facebook oder Twitter. Zusätzlich wird versucht werden, auch nationale Identitätsmanagementlösungen wie beispielsweise die österreichische Bürgerkarte oder den neuen deutschen Personalausweis zu integrieren.

Literatur

[AFGH06] G. Ateniese, K. Fu, M. Green, S. Hohenberger: Improved proxy re-encryption schemes with applications to secure distributed storage. ACM Transactions on Information and System Security (2006), 9(1), 1–30.

[BeTa11] E. Bertino, K. Takahashi: Identity Management: Concepts, Technologies, and Systems. Artech House (2011).

[Cox12] P. Cox: How to Manage Identity in the Public Cloud. InformationWeek (2012).

[Gopa09] A. Gopalakrishnan: Cloud Computing Identity Management. SETLabs Briefings (2009), 7(7), 45–55.

[Goul10] J. T. Goulding: Identity and access management for the cloud: CA's strategy and vision. (2010)

[Hard12] D. Hardt: The OAuth 2.0 Authorization Framework. RFC 6749. (2012)

[HSW+12] D. Hühnlein, J. Schmölz, T. Wich, B. Biallowons, M. Horsch, T. Hühnlein: Standards und Schnittstellen für das Identitätsmanagement in der Cloud. In: DACH Security 2012, syssec (2012) 208-218.

[LCR+08] H. Lockhart, B. Campbell, N. Ragouzis, J. Hughes, R. Philpott, E. Maler, T. Scavo: Security Assertion Markup Language (SAML) V2.0 Technical Overview. (2008).

[NYHR05] C. Neuman, T. Yu, S. Hartman, K. Raeburn: The Kerberos network authentication service (V5). RFC 4120. (2005).

[OpenID] OpenID Foundation: OpenID Authentication 2.0 – Final (2007). http://openid.net/specs/openid-authentication-2_0.html

[PeBe10] S. Pearson, A. Benameur: Privacy, Security and Trust Issues Arising from Cloud Computing. In 2010 IEEE CloudCom (2010). 693–702

[SBJ+14] N. Sakimura, J. Bradley, M. Jones, B. de Medeiros, C. Mortimore: OpenID Connect Core 1.0. (2014).

[Scib06] A. Sciberras: Lightweight Directory Access Protocol (LDAP): Schema for User Applications. RFC 4519. (2006).

[US-PA01] Senate of the United States: Uniting and Strengthening America by Providing Appropriate Tools Required to Intercept and Obstruct Terrorism (USA PATRIOT ACT) Act of 2001.

[ZwZS14] B. Zwattendorfer, T. Zefferer, K. Stranacher: An Overview of Cloud Identity Management-Models. In WEBIST 2014. SCITEPRESS (2014). 82-92