Peter Völkl

Cloud Storage Pool – Virtuelle Vernetzung von Speicherdiensten in der Cloud

109 - Data Science: Erfassung, Modellierung, Analyse und Visualisierung von Daten

Abstract

Es gibt mittlerweile eine Vielzahl an Cloud Services, die auch im Umfeld des Personal Computing Verwendung finden. Cloud Dienste, die inzwischen auch im direkten Umfeld der AnwenderInnen an massiver Bedeutung gewonnen haben, sind Cloud Storages. Dieses Service ermöglicht den AnwenderInnen, ihre Daten online im Cloud Speicher des jeweiligen Serviceproviders abzulegen und weltweit über das Internet darauf zuzugreifen. Es ist jedoch nicht möglich, mehrere Onlinespeicher unterschiedlicher Serviceprovider zu einem integrierten System zu kombinieren. Eine einheitliche Verschlüsselungslösung, die mit allen Cloud Storage Clients gleichermaßen zusammenarbeitet, existiert ebenfalls nicht. Im Rahmen des Forschungsprojektes wurde der Prototyp einer Zwischenschicht zwischen den AnwenderInnen mit ihren Applikationen und den einzelnen Cloud Storage Services entwickelt, welche einen einheitlichen und integrierten Zugriff auf alle angebundenen Datenspeicher ermöglicht. Der Zugriff erfolgt dabei über eine virtuelle Ordnerstruktur, die die einzelnen Speicherdienste integriert. Zusätzlich ist es möglich, einzelne Datenspeicherorte zu verschlüsseln und zu synchronisieren. Für die Clients existiert eine Weboberfläche und eine direkte Anbindung von Anwendungen über eine WebDAV Schnittstelle. Der in dieser Arbeit entwickelte Prototyp ist für die Cloud Speicher Services Dropbox und Google Drive programmiert.

Keywords:

Cloud Storage, Dropbox, Google Drive, Skydrive, Onedrive, WebDAV, IaaS

1. Einleitung

Ein im Umfeld des Personal Computing an Bedeutung gewonnen habender Cloud Dienst ist das Cloud Storage. Die direkten Vorgänger der Cloud Storage Technologien waren die Netzwerkspeicher. Ähnlich dazu wurden für den Zugriff auf Cloud Speicher von den jeweiligen Anbietern eigene Protokolle und Schnittstellen entwickelt. Durch die bessere Skalierbarkeit und Verteilung von Last und Verfügbarkeit kann durch die Verwendung von Cloud Storages oft eine bessere Kosten- und Performanceeffizienz als bei herkömmlichen Speicherlösungen erreicht werden (Ju et al. 2011: 1766-1768).

Die verschiedenen angebotenen Online Speicher können jedoch nicht einfach zu einem integrierten System kombiniert werden. Eine einheitliche Verschlüsselungslösung, die mit allen Clients gleichermaßen zusammenarbeitet, existiert ebenfalls nicht.

Für die Authentifizierung und die damit verbundene Autorisierung werden bei den einzelnen Cloud Storage Anbietern unterschiedliche Verfahren verwendet. Diese können eigene Implementierungen oder allgemein verfügbare Methoden wie beispielsweise OAuth (Messina / Parecki 2012) sein. OAuth ermöglicht die zentrale Verwaltung von BenutzerInnen und Rollen gegenüber Drittanbieter Services. Dabei wird mit Access Token gearbeitet, die vom Autorisierungs-Server an den Client übergeben werden und ihm damit den Zugriff auf die eigentlichen Ressourcen ermöglichen (Babiarz 2014: 4). Bei OAuth sind bisher keine Protokollfehler bekannt, welche von AngreiferInnen ausgenutzt werden konnten. Die Hauptangriffsvektoren sind hierbei fehlerhafte Implementierungen auf Seiten des Services oder des Clients, welche zu Schwachstellen führen können (Babiarz 2014: 13-15).

Die Daten sind innerhalb der verteilten Speicher der Cloud Storage AnbieterInnen meist über Hashwerte adressiert, um sie schnell aufzufinden und Redundanzen innerhalb eines Speicherortes zu vermeiden. Das bedeutet jedoch auch, dass ein Angriff bei Kenntnis des Hashwertes möglich ist. Hier können fehlerhafte Security-Implementierungen der Anbieter ausgenutzt werden. Einen weiteren möglichen Angriffsvektor stellt in diesem Zusammenhang die Ausnutzung des Geburtstagsparadoxons dar. Hierbei kann dem Speicherdienst der Besitz einer Datei durch die Generierung einer anderen Datei mit gleichem Hashwert vorgetäuscht werden (Hash Value Manipulation Angriff). Dieses und ähnliche Angriffsszenarien wurden von Mulazzani et al. (2011) analysiert und anhand des Cloud Storage Services Dropbox getestet und als durchführbar eingestuft.

Neben der besseren Absicherung der Datenübertragungen und der Verifikation des Dateibesitzes (Mulazzani et al. 2011: 9) könnten die Daten mittels Kryptographie auf Dateiebene vor und nach ihrer Übertragung an den Cloud Speicher Service abgesichert werden. Dabei können die schwachen Sicherheitsmechanismen der Anbieter, im schlimmsten Fall, lediglich zum Abruf von verschlüsselten Daten ausgenutzt werden.

Auch die Verfügbarkeit und Integrität der abgelegten Daten ist ein wesentlicher Aspekt der Cloud Security. So sollten kritische und wichtige Daten auf mehrere Speicherorte gespiegelt abgelegt werden, um eine Verfügbarkeit der mit ihnen verbundenen Services zu gewährleisten (Kavitha et al. 2013: 48). Im Falle automatischer Abgleiche muss die Datenintegrität gewährleistet werden, indem mögliche Dateikonflikte bei gleichzeitiger Änderung der Daten auf den einzelnen Spiegelungen erkannt und entsprechend behandelt werden.

Die vorliegende Arbeit fasst das Forschungsprojekt der zugehörigen Master Thesis (Völkl 2014) zusammen, in welcher auch der gesamte Umfang der Analyse und Implementierung nachgelesen werden kann.

2. Zielsetzung

Die Zielsetzung der Arbeit ist es, den Prototyp einer Zwischenschicht zu entwickeln, welcher über eine Weboberfläche und Clientschnittstelle verfügt und eine virtuelle Integration, Spiegelung und Verschlüsselung unterschiedlicher Cloud Storages ermöglicht. Das Anwendungsdesign soll modular erweiterbar gestaltet werden, damit zusätzliche Funktionen auch im Nachhinein ergänzt werden können.

Die AnwenderInnen sollen sich mit ihrem Client direkt beim Cloud Storage Pool anmelden können und danach Zugriff auf die virtuelle Dateistruktur bekommen. Der Aufbau dieser grundlegenden Funktion ist in Abbildung 1 skizziert. Die Ablage und Verteilung der Daten in den verschiedenen Cloud Storages erfolgt dann automatisch im Hintergrund.

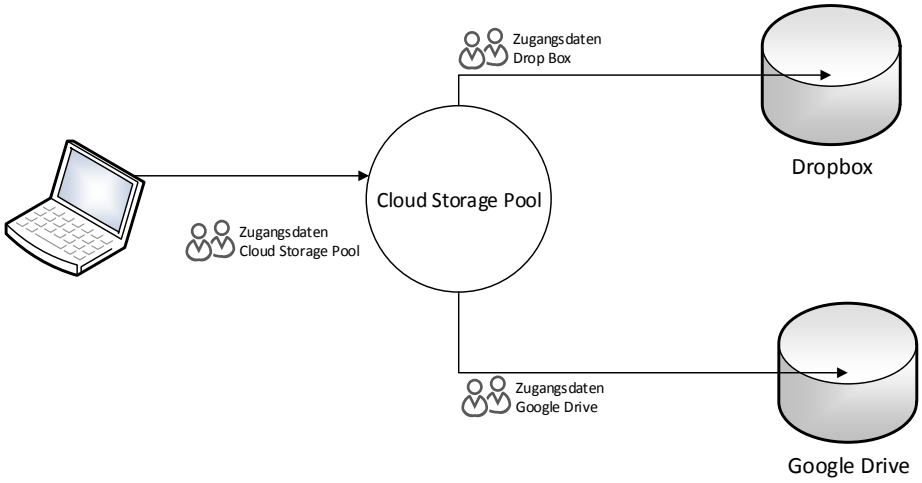

Abbildung 1 - Vernetzung mehrerer Cloud Storage Anbieter durch einen Cloud Storage Pool

3. Vergleich existierender Lösungsansätze

Es wurden bestehende Lösungsansätze der Aufgabenstellung recherchiert und miteinander verglichen. Dabei wurde ein besonderes Augenmerk auf die Datensicherheit und die Unabhängigkeit und Flexibilität der AnwenderInnen gelegt. Die sich daraus ergebenen Anforderungen sind in Tabelle 1 aufgelistet und die einzelnen Lösungen darin gegenübergestellt. Dabei wurden die Systeme OwnCloud, JoliCloud, CloudKafe und ECOCloudS verglichen. OwnCloud (ownCloud Inc. 2014) ist ein Open Source Webservice, mit dem ein eigener Cloud Server betrieben werden kann. Die Entwicklung von ownCloud begann 2010 und wurde seit dem stetig weitergeführt. Jolicloud (Jolicloud 2014) wurde 2009 gegründet und hat seinen Sitz in Paris, Frankreich. Das Hauptprodukt trägt ebenfalls den Namen Jolicloud und kombiniert eine Vielzahl von Cloudservices, wie einen Online Storage Service, Multimediaplayer, Office Anwendungen, Newsfeeds und die Vernetzung von Social Network Accounts, innerhalb eines Internetportals. CloudKafe (CloudKafe 2014) ist ein ähnliches Portal Service wie Jolicloud. Es ermöglicht die Integration mehrerer Cloud Services in eine gemeinsame Oberfläche. Der

Vorteil liegt hier ebenfalls in der gemeinsamen Verwaltung der einzelnen Cloud Speicher und anderer Services, wie Social Networks und Media Services. Bei ECOCloudS (Hausdorf 2013) handelt es sich um eine Clientapplikation, welche den Clientzugriff und die Ordnersynchronisation mehrerer Cloud Storages kombinieren kann. Die Anwendung befindet sich im Stadium eines Prototyps, verfolgt jedoch sehr vielversprechende Ansätze zu Vernetzung und Synchronisation unterschiedlicher Cloud Storage Services sowie zu deren Datensicherheit durch Verschlüsselung und Speichereffizienz durch Komprimierung.

Keine der bisher verfügbaren Lösungen konnte alle gestellten Anforderungen des Vergleichs erfüllen. Jede hat damit einen speziellen Nutzen für nur einige der möglichen Anwendungsfälle. Die AnwenderInnen müssen sich dabei zwischen Flexibilität und Datensicherheit entscheiden und im Falle einer gewünschten Synchronisation auf die Anbindung mehrerer Arbeitsplätze verzichten, da ansonsten die Datenintegrität gefährdet würde. Die Entwicklung eines neuen Lösungsansatzes, welcher alle Anforderungen erfüllt, ist daher zielführend.

Tabelle 1 - Vergleich bisher verfügbarer Lösungsansätze

Kriterium	OwnCloud	Jolicloud	CloudKafe	ECOCloudS
Private Installation auf eigenem System möglich?	Ja	Nein	Nein	Ja
Weboberfläche vorhanden?	Ja	Ja	Ja	Nein
Clientsoftware vorhanden?	Ja	Nein	Nein	Ja
Schnittstelle zur Integration in eigene Software vorhanden?	Ja	Nein	Nein	Nein
Darstellung für AnwenderInnen als integrierter Datenspeicher?	Ja	Nein	Ja	Ja
Synchronisation zwischen den Datenspeichern möglich?	Nein	Nein	Nein	Ja, jedoch mögliche Probleme der Integrität
Verschlüsselung der Datenspeicher möglich?	Nein	Nein	Nein	Ja

4. Implementierung des Prototyps und Ergebnis

Die einzelnen Cloud Datenspeicher wurden in Form einer virtuellen Ordnerstruktur mit dem Namen „Cloud Storage Pool“ (CSP) abgebildet, wobei sich je ein Ordner eines externen Datenspeichers inklusive seiner Dateien und Unterordner in mindestens einem virtuellen Ordner des CSP befindet. Für den Zugriff über die Endgeräte der AnwenderInnen wurde eine Weboberfläche und eine WebDAV

Schnittstelle implementiert, um den Zugriff mit nahezu jedem internetfähigen Endgerät und eine Integration in bestehende Dateistrukturen auch betriebssystemunabhängig zu ermöglichen.

Der Aufbau der virtuellen Ordnerstruktur ist in Abbildung 2 skizziert. Der Zugriff des Clients erfolgt zentral. Die virtuellen Ordner sind mit den jeweiligen Cloud Storages verbunden und können auch verschlüsselt oder serviceübergreifend synchronisiert werden.

Um die einzelnen virtuellen Ordner miteinander abzugleichen, wurde die Möglichkeit von automatischen Spiegelungen beliebig vieler externer Datenspeicher entwickelt. Die Synchronisation erfolgt über die Informationen der jeweiligen Cloud Storage Services. Dazu wird im CSP für jeden verbundenen virtuellen Ordner eine Dateiliste mit den zugehörigen Änderungen abgerufen. Die Datenspeicher werden dabei immer als primärer und als sekundärer Datenspeicher betrachtet, um mögliche Konflikte behandeln zu können.

Zur Gewährung der Vertraulichkeit der in den Cloud Speichern abgelegten Daten wurde eine synchrone Verschlüsselung implementiert, um die Daten schnell und einfach verschlüsselt ablegen und entschlüsselt abrufen zu können. Die Verschlüsselung ist dabei für jeden virtuellen Datenspeicher separat aktivierbar und kann auch mit unterschiedlichen Schlüsseln versehen werden.

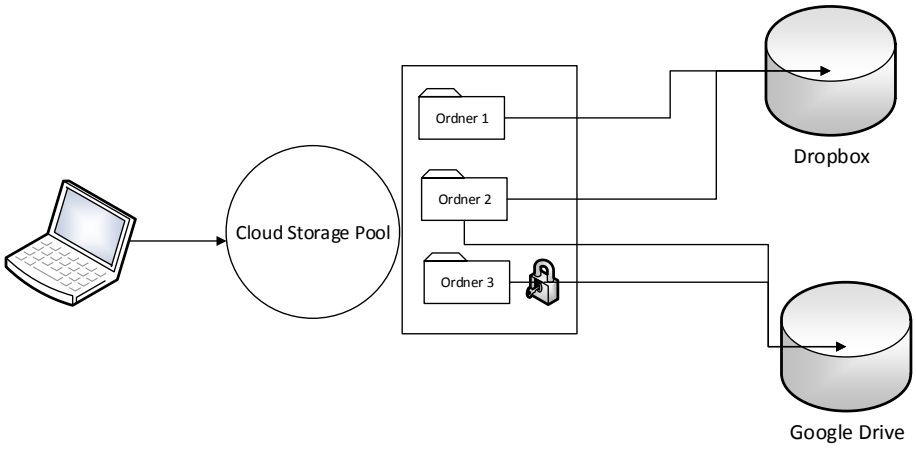

Abbildung 2 - Virtuelle Ordnerstruktur des Cloud Storage Pools mit unterschiedlichen Cloud Storages

Im Rahmen des Prototyps erfolgt eine Anbindung der Cloud Storage Dienste Dropbox und Google Drive, da es sich dabei um die beiden bei den EndanwenderInnen am weitesten verbreiteten Dienste handelt. Die Implementierung des Prototyps erfolgte in der Programmiersprache PHP und unterstützt damit die serverseitig im Web am meisten verbreitete (Q-Success 2015) Laufzeitumgebung.

Die Klassenstruktur des CSP besteht aus einer Klasse für den WebDAV Zugriff und den Klassen für die Verbindung der externen Cloud Datenspeicher in die virtuelle Ordnerstruktur. Da die externen Datenspeicher in Form einer virtuellen Ordnerstruktur in den CSP eingebunden werden sollen, wird

hierfür eine abstrakte Klasse verwendet, welche die allgemeingültige Struktur für die Zugriffe der CSP Anwendungen vorgibt. Aus ihr werden wiederum die Klassen für den Zugriff auf den jeweiligen Cloud Speicherdienst abgeleitet und die abstrakten Objektmethoden fertig implementiert.

Da der Datenzugriff unterschiedlicher Speicherdienste nicht immer in einer regulären Dateibaumstruktur erfolgen muss, kann es erforderlich sein, eine entsprechende Umformung der Zugriffe vorzunehmen. Eine solche Anpassung war auch im Prototyp nötig, da Google Drive keinen direkten Zugriff auf Dateipfade ermöglicht. Dazu wurden hier die Klassenmethoden getMetadataByPath und getPathById implementiert, welche den benötigten Dateipfad aufgrund der Dateiverknüpfungen in Google Drive ermitteln. Die Funktionsweise der Pfadzugriffe über die Objektmethoden getMetadataByPath und getPathById sind in Abbildung 3 skizziert.

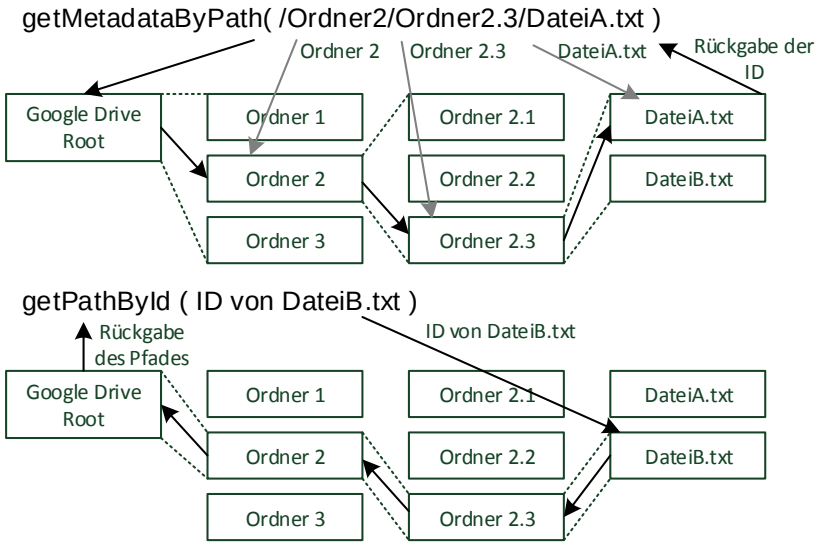

Abbildung 3 - Objektmethoden für den Google Drive Zugriff über Dateipfade

Die Methoden zur Ver- und Entschlüsselung der Dateiübertragungen sind bereits in der abstrakten Klasse fertig programmiert und können gegebenenfalls auch in den abgeleiteten Klassen überlagert werden, da sie nicht als final deklariert sind und eine Überlagerung zur Änderung des Kryptographie-Verfahrens einzelner externer Datenspeichertypen durchaus notwendig sein kann.

Eine Anbindung an weitere Speicherdienste ist durch weitere Implementierungen von Ableitungen der abstrakten Klasse jederzeit möglich. Welche Art von Speicherdienst sich darin befindet ist für die restliche Anwendung irrelevant. Es wäre daher auch möglich, Zugriffe auf Datenbankspeicher oder lokale Daten zu integrieren.

Der Quellcode des Prototyps wurde im Filehosting-Dienst für Software-Entwicklungsprojekte GitHub veröffentlicht[1].

[1] Cloud Storage Pool auf Github: https://github.com/petervoe/Cloud-Storage-Pool

5. Diskussion und Ausblick

Der entwickelte Prototyp CSP deckt alle gestellten Anforderungen ab, um einen zentralen Zugriff auf mehrere unterschiedliche Datenspeicher in der Cloud und die Gewährung der Datensicherheit zu ermöglichen. Dies sind insbesondere die transparente und integrierte Darstellung der Ordnerstruktur und die einheitliche Ablage von Dateien und Ordnern. Die Ver- und Entschlüsselung der Dateiübertragungen ermöglicht eine sichere Ablage von vertraulichen Informationen in der Cloud, ohne einen zusätzlichen Aufwand für die AnwenderInnen zu verursachen. Auch die Anbindung weiterer Speicherdienste ist jederzeit möglich.

Zukünftig ist eine Weiterentwicklung des CSP Prototyps als Integration in Web-Portalanwendungen und lokale Netzwerkgeräte wie Network Attached Storages (NAS) und All-in-One Router denkbar, um EndanwenderInnen einen eigenen und leicht zu konfigurierenden lokalen Betrieb des CSP zu ermöglichen.

Literaturliste/Quellenverzeichnis:

Babiarz, P. (2014): Sicherheitsauswirkungen von verknüpften Cloud-Diensten. Wien: Fachhochschule Technikum Wien.

CloudKafe (2014): CloudKafe - Organizing your Cloud!. https://www.cloudkafe.com/, (18.04.2014)

Hausdorf, M. (2013): Cloud-Speicher sicher und effizient nutzen. Wien: Fachhochschule Technikum Wien.

Jolicloud (2014): Jolicloud. http://www.jolicloud.com/, (18.04.2014)

Ju, J./Jiyi, W./Jianqing, F./Zhijie, L./ Jianlin, Z. (2011): A Survey on Cloud Storage. In: Journal of Computers 6, 1746-1771.

Kavitha, M. G./Vinay Kumar, N. A./Balasubrhamanya (2013): Secure Cloud Storage with Multi Cloud Architecture. In: International Journal of Innovative Technology and Exploring Engineering 3, 45-49.

Messina, C./Parecki, A. (2012): OAuth Community Site. http://oauth.net/, (15.01.2016)

Mulazzani, M./Schrittwieser, S./Leithner, M./Huber, M./Weippl, E. (2011): Cloud Speicherdienste als Angriffsvektoren. In: Proceedings of 9. Sicherheitskonferenz Krems, 25-38.

ownCloud Inc. (2014): ownCloud. http://owncloud.org/, (18.04.2014)

Q-Success (2015): Usage Statistics and Market Share of Server-side Programming Languages for Websites, January 2016. http://w3techs.com/technologies/overview/programming_language/all/, (15.01.2016)

Völkl, P. (2014): Cloud Storage Pool - Virtuelle Vernetzung von Speicherdiensten in der Cloud. Wien: Ferdinand Porsche FernFH.