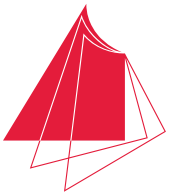

Hochschule Karlsruhe
Technik und Wirtschaft
UNIVERSITY OF APPLIED SCIENCES

Analyse einer Referenzimplementierung des ActivityPub Standards mit Blick auf die Sicherheit

Bachelor Thesis von

Armin Kunkel

an der Fakultät für Informatik und Wirtschaftsinformatik
Fachrichtung Verteilte Systeme (VSYS)

Erstgutachter: Prof. Dr. rer. nat. Christian Zirpins
Betreuer: M. Sc. Robert Schäfer

01. Dezember 2018 – 31. März 2019

Hochschule Karlsruhe Technik und Wirtschaft
Fakultät für Informatik und Wirtschaftsinformatik
Moltkestr. 30
76133 Karlsruhe

Ich versichere wahrheitsgemäß, die Arbeit selbstständig angefertigt, alle benutzten Hilfsmittel vollständig und genau angegeben und alles kenntlich gemacht zu haben, was aus Arbeiten anderer unverändert oder mit Änderungen entnommen wurde.

Karlsruhe, 19. Dezember 2018

. . . . . .

(Armin Kunkel)

Zusammenfassung

Deutsche Zusammenfassung

Inhaltsverzeichnis

Abbildungsverzeichnis

Tabellenverzeichnis

1. Einleitung

Zentralisierung von Daten und das Vertrauen auf einzelne Instanzen ist heutzutage allgegenwärtig. Im Internet gibt es haufenweise soziale Netzwerke wie Facebook, Twitter, Google+, Instagram oder Pinterest die zumeist das Recht an den Daten, die Sie von ihren Nutzer bekommen, behalten und nicht zuletzt diese Daten auch verkaufen für zB. Werbezwecke. Für Cracker ist es außerdem leichter an die gesamte Datenbank zu kommen, da bei zentralisierten Netzwerken meist auch eine zentrale Datenbank, bzw. ein zentraler Zugriffspunkt, vorhanden ist. Wenn der Zugriff auf diesen/diese erreicht ist kann die ganze Datenbank ausgelesen werden.

Was bedeutet nun dezentral? In der Politik bedeutet Dezentralisierung „die Übertragung zentralstaatlicher Aufgaben auf subnationale oder subsidiäre Ebene(n)“[**wikipedia-dezentralisier**
In der Energiewirtschaft spricht man von einer „dezentralen Stromerzeugung“, wenn der Strom an den Stellen wo er verbraucht auch erzeugt wird. Ein Beispiel hierfür wäre ein Wasserkraftwerk, dass den Strom für die umgebenen Dörfer oder Städte liefert[**wikipedia-dezentralisie**
Dezentralisierung in der Informatik bedeutet das vermaschte Vernetzen von Computern untereinander[**wikipedia-dezentralisierung-informatik**].

Durch die Verteilung der zentralstaatliche Aufgaben auf subnationale oder subsidiäre Ebene wird die zentralstaatliche Ebene entlastet und hat somit mehr Ressourcen für andere Aufgaben. Beim Errichten von Wasserkraftwerken nahe an Dörfern und Städten entfällt der Transport des Stroms und somit verringert sich der Verlust beim Transport über das Stromnetz. Das Dezentralisieren von sozialen Netzwerken bring verschiedene Vorteile mit sich. Es wird zum Beispiel möglich, die einzelnen Instanzen selbst zu moderieren und somit die Kontrolle zu dezentralisieren oder Inhalt aus einem sozialen Netzwerk heraus in einem anderen zu erstellen. Bei ActivityPub wird dies über die Förderierung gewährleistet.

ActivityPub besteht aus zwei Teilen. Dem Client-zu-Server Protokoll und dem förderierten Server-zu-Server Protokoll. Durch die ActivityPub W3C Empfehlung Anfang 2018 und die Social-Web-Working-Group Arbeitsgruppe des W3C sowie weiteren, wird die Verbreitung des Protokolls vorangebracht, vermutlich um den Netzwerkeffekt nutzen zu können. Der Netzwerkeffekt ist ein aus der Volkswirtschaftslehre stammender Begriff. Auf Übertragen bedeutet er soviel wie: SSobald die kritische Masse an Netzwerken erreicht wurde, tritt der positive Feedback-Effekt auf, sodass der Nutzen für Netzwerke, welche den Standard implementieren, steigt".

Bibliothek gehen(Quellen): Netzwerkeffekt, Dezentralisierung 3 mal

Um die Arbeit erfolgreich abzuschließen soll eine Implementierung des verteilten Server-zu-Server Protokolls als Prototyp und eventuell ein Konzept zum sicheren Einsatz des Protokolls vorliegen. Darüber hinaus soll die Arbeit einen Überblick über die Grundlagen des Protokolls geben und wie die relevanten Bestandteile zu verstehen sind. In diesem Zusammenhang bedeutet „sicher“, dass sichere Authentifizieren vom Client gegenüber dem Server und zusätzlich der Server untereinander, sowie das Sicherstellen der manipulationsfreien Übertragung der Inhalte.

Der ActivityPub[**activityPub**] Standard wurde am 23 Januar 2018 von der W3C empfohlen und von einer Arbeitsgruppe des W3C, der Social Web Working Group(SWWG)[**socialWg**, **pushSocialWeb**], entwickelt. Diese Gruppe war vom 21. Juli 2014 bis zum 13 Februar 2018 aktiv[**socialWg**] und entwickelte unter anderem ActivityPub, Activity Streams Core[**activityStreamsCore**], Activity Streams Vocab[**activityStreamsVocabulary**]. Ziel der SWWG war es neue Protokolle, Vokabulare und API's zu definieren für den Zugriff auf soziale Inhalte der sogennanten Open Web Platform[**social-wg-charter**].

1.1. Motivation

Bei zentralen sozialen Netzwerken, z.B. Facebook, besitzt die Kontrolle über die Daten das Unternehmen. Durch die Dezentralisierung bleiben die Daten bei den einzelnen Instanzen des sozialen Netzwerkes. Die Datenbanken bei zentralen Netzwerken können zwar verteilt sein und auch das Netzwerk könnte dezentral angelegt sein, trotzdem bleiben die Daten bei einem Unternehmen. Außerdem können bei zentralen sozialen Netzwerken nicht ohne weiteres Inhalte mit anderen sozialen Netzwerken Ausgetauscht werden. Mit einem Protokoll wie ActivityPub wird das verteilen sowie der Zugriff auf Inhalte und der Austausch von Aktivitäten über mehrere soziale Netzwerke hinweg ermöglicht.

Protokolle wie „Diaspora Federation und OStatus“ bieten den Zugriff auf dezentral gespeicherte soziale Inhalte und ermöglichen das veröffentlichen von sozialen Inhalten. Der im März letzten Jahres vom W3C empfohlene Standard namens „ActivityPub“ wurde auf Basis des Wissens im Umgang mit dem OStatus und Pump.io Protokoll entwickelt[**activityPub-acknowledg**
ActivityPub ist wie „OStatus“ ein Protokoll für dezentrale soziale Netzwerke und wird in dieser Bachelor Arbeit untersucht.

2. Grundlagen zur sicheren Umsetzung dezentraler sozialer Netzwerke

„ActivityStreams Core“ ActivityStreams Core beinhaltet Modelle für Aktoren, Aktivitäten, Intransitiven Aktivitäten, Objekte, Links, Sammlungen, Natürliche Sprachwerte (Strings) und für Internationalisierung.

2.1. ActivityPub Standard

ActivityPub definiert zwei Protokollschichten, sowie Konzepte, Sammlungen und Interaktionen für dezentrale soziale Netzwerke. Eine Protokollschicht ist das Client-zu-Server Protokoll, um Clients den Zugriff auf einen Server zu ermöglichen sowie zum entgegennehmen von Anfragen. Die zweite Protokollschicht besteht aus dem förderierten Server-zu-Server Protkoll, welches den einzelnen Instanzen eines dezentralen sozialen Netzwerkes den Austausch von Inhalten untereinander gestattet. ActivityPub setzt auf bereits bestehende Empfehlungen des W3C auf, welche teilweise auch von der SWWG entwickelt wurden wie zB. Activity Streams Core und Activity Streams Vocab.

Auch andere Technologien wie JSON-LD[**jsonLd**] werden benutzt um die Erweiterbarkeit zu gewährleisten. Über neue Ontologien (Vokabulare) können weitere syntaktische Definitionen und semantische Beschreibungen zu den bestehenden hinzugefügt werden. Diese Vokabulare können im Kontext des JSON-LD Objektes, also der Aktivität, angegeben werden.

(Stimmt das mit den Ontologien???) (Beispiel Context Abbildung mit erweitertem Context???)

2.1.1. Bestandteile des Protokolls

Im Grunde besteht der ActivityPub Standard aus zwei Protokollen:

- ActivityPub Client-zu-Server Protokoll (Social API)
- ActivityPub verteiltes Server-zu-Server Protokoll (Federation Protocol)

Die beiden Protokolle können unabhängig voneinander implementiert werden. Das Client-zu-Server Protokoll besteht aus einem Client und Server Teil.

In ActivityPub werden Benutzer als „Aktoren“(actors) dargestellt. Jedes Aktoren Objekt

muss eine „Inbox“ und „Outbox“, welche geordnete Sammlungen sein müssen, sowie eine ID und ein Typ besitzen[**activityPubActor**]. Die ID muss global einzigartig sein und der Typ kann variieren zwischen „Person“, „Application“ und weiteren.[**activityStreamsCoreActor**]

Listing 2.1: Beispiel Aktoren Objekt

2.1.2. Zugehörige Standards

ActivityPub benutzt die ActivityStream Daten Syntax und das Vokabular. Zusätzlich kann das Sicherheitsvokabular von Webpayments benutzt werden um Syntax und Semantik für das Beschreiben eines öffentlichen Schlüssels o.ä. zur Verfügung zu stellen.

2.2. Client-zu-Server Kommunikation

Der Client empfängt Nachrichten über zwei Schnittstellen. Er kann sich per HTTP GET Anfrage auf seine eigene „Inbox“ die neusten an ihn adressierten Inhalt holen. Außerdem kann eine HTTP GET Anfrage an die „shared Inbox“ des Servers gesendet werden um eine Repräsentation aller Inhalte der Instanz zu bekommen.
(Oder alle Inhalte des bekannten Netzwerks???)

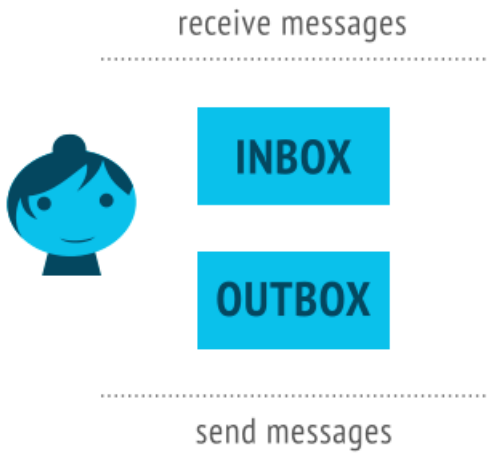

Abbildung 2.1.: Interaktionen des Client mit dem Server

Um neue Aktivitäten an seine Folgenden, direkt an einzelne Personen oder sichtbar innerhalb der Domain zu veröffentlichen kann der Client eine HTTP POST Anfrage an seine „Outbox“ senden mit einer Aktivität oder einem ActivityStreams 2.0(AS2) Objekt als Inhalt.

2.2.1. Client Teil

Der Client Teil des Client-zu-Server Protokolls beinhaltet mehrere Aufgaben:

- Auffinden der URL über das Profil
- Serialisieren der Daten in eine Aktivität oder AS2 Objekt
- Adressieren von Aktivitäten oder AS2 Objekten
- Eine HTTP POST Anfrage mit „Content-Type: application/ld+json: profile="https://w3.org/ns/activ „Content-Type: application/activity+json: profile="https://w3.org/ns/activitystreams"“ an seine „Outboxg“ absetzen
- Bei den Aktivitätstypen CREATE, UPDATE, DELETE, FOLLOW, ADD, REMOVE, LIKE, BLOCK, UNDO wird die „object“ Eigenschaft bereitgestellt
- Die „target“ Eigenschaft wird bei den Aktivitätstypen ADD und REMOVE bereitgestellt
- Anfragen müssen mit den Zugangsdaten des Nutzer, zu dem die „Outbox“ gehört, authentifiziert werden

Zusammengefasst muss sich der Client um das Entdecken von Endpunkten - Inbox/Outbox - über Profile, das Serialisieren von Daten zu Aktivitäten oder AS2 Objekten sowie um das setzen der entsprechenden Kopfzeilen (Mittlerer Teil der obigen Auflistung) und absenden der HTTP POST Anfrage kümmern. Außerdem muss bei HTTP GET Anfragen an den Server die „ACCEPT: application/ld+json“Kopfzeile gesetzt sein.

2.2.2. Server Teil

Zu den Aufgaben des Serverteils gehören das Annehmen von HTTP POST Anfragen auf die „Outbox“ eines Nutzers und das verifizieren ob der Nutzer berechtigt ist diese Anfrage zu tätigen. Weiter werden die „Inbox“ Endpunkte bereitgestellt um authentifizierten Nutzern den Zugriff auf die neusten Inhalte zu geben.

(Ist das richtig oder bestehen die Aufgaben des Serverteils auch im weiterleiten zu anderen Instanzen, weil das ist ja eigentlich der Server-zu-Server Teil???) (Ist der Server Teil ein interner Speicher Mechanismus der Instanz oder ein delivery Mechanismus für alle Instanzen des Netzwerk oder für alle Instanzen mit einer Federated Implementierung insgesamt??? Ich verstehe es wie oben beschrieben, also ohne Server-to-Server Teil alles innerhalb einer Instanz). Angenommen der Server Part beinhaltet nicht nur das annehmen und abspeichern innerhalb einer Instanz, sonder auch das weiterleiten an andere Instanzen: Dann müsste ein Nutzer alleine durch die Implementierung des Server Teils des Client-zu-Server Protokolls in der Lage sein Inhalt mit anderen Servern auszutauschen. Den Unterschied zu einem förderierten Server liegt dann im Empfangen von öffentlichen Nachrichten (5.6 Public Addressing | Shared Inbox Delivery) und dem Weiterleiten (7.1.2 Forwarding from Inbox) von auf dieser Domäne nicht zustellbaren Empfängern. Im Endeffekt ist die Frage ob der Server Teil des Client-zu-Server Protokolls eine Teilmenge des Server-zu-Server Protokolls ist???

2.3. Server-zu-Server Kommunikation

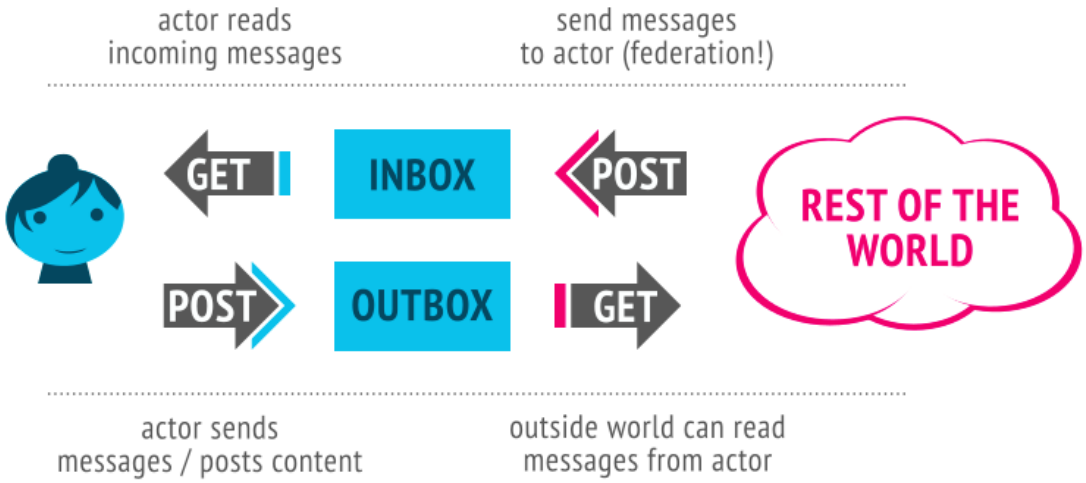

Abbildung 2.2.: Schnittstellen des ActivityPub Protokolls

Bei der Server-zu-Server Kommunikation bestehen die Hauptaufgaben im annehmen und zustellen von HTTP POST Anfragen auf die „Inboxen“ und „Outboxen“ der Nutzer und im weiterleiten von Aktivitäten die der Server nicht zustellen kann. Dazu kommt das Bereitstellen der „Outbox“ Sammlungen.

2.4. Authentifizierung und Datenintegrität

Für die Authentifizierung und zum sichern der Datenintegrität definiert der Standard keine Mechanismen. Es gibt allerdings „Best Practices“ für die Umsetzung dieser Anforderungen.

Zum einen werden bei der Client-zu-Server Authentifizierung „OAuth 2.0“ Tokens benutzt, zum anderen auf der Server Seite HTTP Signaturen oder „Linked Data Signatures“zur Sicherstellung der Datenintegrität. Letztere Signatur wird eher verwenden wenn Objekte übertragen und weitergeleitet werden, da damit die Integrität des Objekts an sich verifiziert werden kann.[1]

[1] https://www.w3.org/wiki/SocialCG/ActivityPub/Authentication_Authorization

3. Analyse bestehender Ansätze

3.1. Vergleich bestehender Implementierungen

3.2. Verwandte Protokolle

- OStatus
- Diaspora Federation

4. Entwurf einer Lösung für sichere Server-zu-Server Interaktion mit ActivityPub

4.1. Entwurfsentscheidung

4.2. Technische Architektur

5. Implementierung eines ActivityPub Prototyps

5.1. Server-zu-Server Protokoll

6. Evaluation

6.1. Anwendungsbeispiel

6.2. (Performanzmessungen)

6.3. Disskusion von Vor- und Nachteilen der Lösung

7. Zusammenfassung und Ausblick

A. Anhang

A.1. First Appendix Section

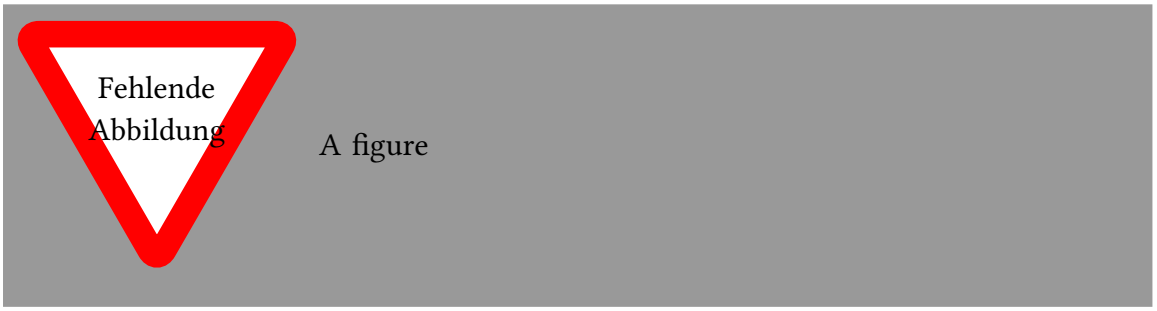

Abbildung A.1.: A figure

...