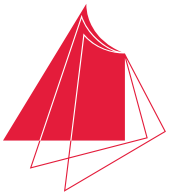


Hochschule Karlsruhe
Technik und Wirtschaft
UNIVERSITY OF APPLIED SCIENCES

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, 30. Dezember 2018

. . . . . .

(Armin Kunkel)

Zusammenfassung

Deutsche Zusammenfassung

Inhaltsverzeichnis

1. Einleitung

Zentralisierung von Daten und das Vertrauen auf einzelne Instanzen ist heutzutage allgegenwärtig. Im Internet gibt es eine Fülle an sozialen Netzwerken 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 Kriminelle sowie Geheimdienste 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.

Bibliothek gehen(Quellen): Netzwerkeffekt, Dezentralisierung 3 mal

Ziel der Arbeit ist es eine Implementierung des verteilten Server-zu-Server Protokolls als Prototyp und ein Konzept zum sicheren Einsatz 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“, das sichere Authentifizieren vom Client gegenüber dem Server und zusätzlich der Server untereinander, sowie das Sicherstellen der manipulationsfreien Übertragung von Inhalten.

@Zirpins Was genau soll im oberen Absatz weiter erklärt werden???

@Robert Laut "wissenschaftlichen Arbeiten"Buch sollte die Geschichte und Stand der Forschung in die Einleitung rein und die Einleitung sollte nicht länger als 1 Seite sein. Ich werde den Teil deshalb mal hier belassen. Außerdem hab ich den 2. und 3. Absatz in die Motivation verschoben

Der ActivityPub[**activityPub**] Standard wurde am 23 Januar 2018 von der WWW Consortium (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, ActivityStreams Core[**activityStreamsCore**], ActivityStreams Vocab[**activityStreamsVocabular**
Die SWWG war eine Arbeitsgruppe des W3C mit dem Ziel neue Protokolle, Vokabulare und API's zu definieren für den Zugriff auf soziale Inhalte der sogenannten Open Web Platform (OWP)[**social-wg-charter**].

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 SWWG Arbeitsgruppe des W3C sowie weiteren, wird die Verbreitung des Protokolls vorangebracht.

Der Netzwerkeffekt ist ein aus der Volkswirtschaftslehre stammender Begriff. Als Beispiel sei das Telefonnetz genannt. Umso mehr Leute ein Telefon besitzen, umso höher ist der Nutzen für jeden einzelnen Telefon Besitzer. Übertragen auf ActivityPub bedeutet das soviel wie: „Je mehr Netzwerke den Standard implementieren, desto höher ist der Nutzen für ein einzelnes Netzwerk und im Endeffekt für die einzelnen Nutzer“.

1.1. Motivation

Bei zentralen sozialen Netzwerken, z.B. Facebook, besitzt die Kontrolle über Daten das Unternehmen. Durch die Dezentralisierung bleiben 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 gehören die Daten einem Unternehmen. Außerdem können bei zentralen sozialen Netzwerken nicht ohne weiteres Inhalte mit anderen 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.

Was bedeutet nun dezentral? In der Politik bedeutet Dezentralisierung „die Übertragung zentral staatlicher Aufgaben auf subnationale oder subsidiäre Ebene(n)“[**wikipedia-dezentralisierung**
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-dezentralisierun**
In der Informatik versteht man unter Dezentralisierung das verteilen von u.A. Daten über mehrere unabhängige Server hinweg.

Durch die Verteilung der zentral staatliche Aufgaben auf sub-nationale oder subsidiäre Ebene wird die zentral staatliche Ebene entlastet und hat somit mehr Ressourcen für andere Aufgaben. Beim Errichten von Wasserkraftwerken nahe an Dörfern und Städten entfällt der Transport des Stroms über größere Distanzen und somit verringert sich der Verlust beim Transport über das Stromnetz. Dezentralisieren von sozialen Netzwerken bring verschiedene Vorteile mit sich. Ein Vorteil ist die Skalierbarkeit bei dezentralen sozialen Netzwerken. Durch das hinzufügen weiterer Instanzen können dem Netzwerk neue Ressourcen bereitgestellt werden. Ein weiterer Vorteil liegt darin, dass keine zentrale Kontrollinstanz vorhanden ist welche korrumpiert werden kann, sei es durch Kriminelle, wirtschaftliche Interessenverteter oder staatliche Autoritäten.

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**

Quelle für dezentralen Zugriff

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

Einordnung des Kapitels + Übersicht über Unterkapitel

Allgemeine Grundlagen zu Sozialen Netzwerken/Social Media und deren Sicherheitsaspekte

Welche Klassen von Anwendungen und Implementierungen gibt es allgemein bei sozialen Netzwerken? Einordnen von ActivityPub!!

AP Beschreibung in hinteren Teil des Grundlagenkapitels oder zu Beginn des 3. Kapitels

„ActivityStreams 2.0“ beinhaltet Modelle für Aktoren, Aktivitäten, Intransitiven Aktivitäten, Objekte, Links, Sammlungen, Natürliche Sprachwerte (Strings) und für Internationalisierung. Das Kernvokabular von ActivityStreams 2.0 wird durch ActivityStreams Vocab erweitert. Dazu gehören verschiedene Aktivitätstypen wie z.B. „Accept“,„Add“,„Remove“,„Delete“ und „Create“[1], um Aktorentypen wie „Person“, „Application“ und „Group“[2] sowie um verschiedenste Objekttypen wie „Article“, „Event“, „Note“ und „Relationship“[3].

Sammlungen beschreiben!

@Alle Wusste eben nicht wie ich das anders machen soll außer als Fußnote. Wie könnte ich ein ßiehe Weblink"gestalten??? Weil alles hier nochmal aufzulisten wäre denke ich nicht gut..

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 (Social API), 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 (Federation Protocol), welches den einzelnen Instanzen von dezentralen sozialen Netzwerken den Austausch von Inhalten untereinander gestattet. ActivityPub setzt auf bereits bestehende Empfehlungen des W3C auf, welche teilweise

[1] activity-types https://www.w3.org/TR/activitystreams-vocabulary/
[2] actor-types https://www.w3.org/TR/activitystreams-vocabulary/
[3] object-types https://www.w3.org/TR/activitystreams-vocabulary/

auch von der SWWG entwickelt wurden wie zB. ActivityStreams Core und ActivityStreams Vocab.

Auch andere Technologien wie JSON Linked Data werden benutzt um die Erweiterbarkeit zu gewährleisten. Über neue Ontologien (Vokabulare) können weitere syntaktische Definitionen und semantische Beschreibungen zu den bestehenden hinzugefügt werden[**activityPub**]. Diese Vokabulare können im Kontext des JSON Linked Data Objektes, angegeben werden. Bei ActivityPub wird das ActivityStreams 2.0 Vokabular verwendet welches durch ActivityStreams Vocab erweitert wird.

2.1.1. Bestandteile des Protokolls

Das Client-zu-Server sowie förderierte Server-zu-Server Protokoll können unabhängig voneinander implementiert werden. Ersteres besteht aus einem Client und Server Teil.

In ActivityPub werden Benutzer als „Aktoren"(actors) dargestellt. Diese können nicht nur Personen, sondern auch Applikationen, Organisationen, Gruppen und Services sein[**activityStreamsCo**
Jedes Aktoren Objekt muss eine „Inbox" und „Outbox", welche geordnete Sammlungen sein müssen, sowie eine ID und ein Typ besitzen[**activityPub**]. Die ID muss global einzigartig sein. Dies kann garantiert werden durch eine Domänen und Protokoll bezogene URI oder IRI wie zum Beispiel „https://example.org/users/alice"oder „https://example.org/alice/áŷýà/2". Der Typ eines Aktor (z.B. "type": "Create") kann variieren zwischen den fünf oben genannten.

2.1.2. Zugehörige Standards und Komponenten

ActivityPub benutzt die ActivityStreams Daten Syntax und das Vokabular. Zusätzlich kann ein weiteres Sicherheitsvokabular[4] benutzt werden um Definitionen zum Bereitstellen eines öffentlichen Schlüssels, Signaturen sowie Verschlüsselten Inhalten u.v.m. zu haben. Am 22 April 2016 hat die „W3C Community Group" einen Entwurfsbericht herausgebracht. Durch diesen wird neue Syntax und Semantik definiert um Internet basierten Applikationen das Verschlüsseln, Entschlüsseln sowie digitale signieren und verifizieren von verlinkten Daten (Linked Data) zu ermöglichen. Es enthält auch Vokabeln für die Erstellung und Verwaltung einer dezentralen Public-Key-Infrastruktur über das Internet[**security-vocab-linked-data**]. Ein Anwendungsfall ist das holen des öffentlichen Schlüssels eines Nutzers, über dessen Aktoren Objekt, um eine von Nutzer gesendete Nachricht zu verifizieren.

JSON Linked Data ist eine Erweiterung des JSON Formates um verlinkte Daten zu Repräsentieren. JSON an sich, ist ein Format welches im Web häufig Anwendung findet um Daten auszutauschen. Im Kern sind ActivityStreams 2.0 auch JSON Linked Data Objekte. Der ActivityStreams 2.0 Kontext definiert verschiedene Klassen und Eigenschaften, von

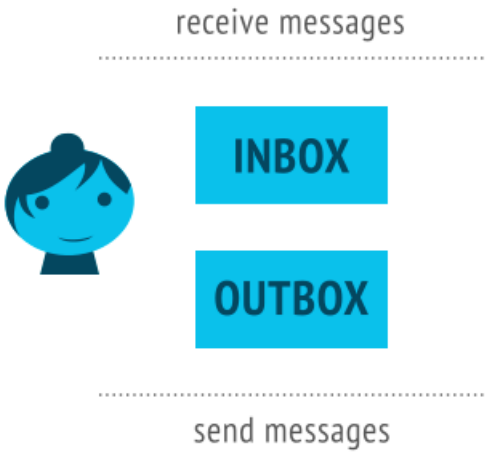
[4]Eine Ontologie die Sicherheitsaspekte definiert wie öffentliche Schlüssel, Signaturen u.v.m.

denen nicht alle benutzt werden. Typische Klassen sind „Activity“, „Link“ und „OrderedCollection“. Ein Beispiel ActivityStreams 2.0 Objekt sieht wie folgt aus:

Listing 2.1: Beispiel ActivityStreams 2.0 Objekt

2.2. Client-zu-Server Kommunikation

Der Client interagiert über zwei Schnittstellen mit dem Server. Er kann sich per HTTP GET Anfrage auf seine eigene „Inbox“ die neusten an ihn adressierten Inhalte holen und über eine HTTP POST Anfrage auf seine „Outbox“ neue Inhalte erstellen.



Quelle: ActivityPub 2018 - Overview

Abbildung 2.1.: Interaktionen des Client mit dem Server

Bildquelle richtig angeben..

Um neue Aktivitäten an seine Folgenden, direkt an einzelne Personen, sichtbar innerhalb der Domain oder unangemeldet für alle zugänglich zu veröffentlichen, kann der Client eine HTTP POST Anfrage an seine „Outbox“ senden mit einer Aktivität oder einem ActivityStreams 2.0 Objekt als Inhalt. Der Client muss mehrere Aufgaben, wie in **??** beschrieben, übernehmen.

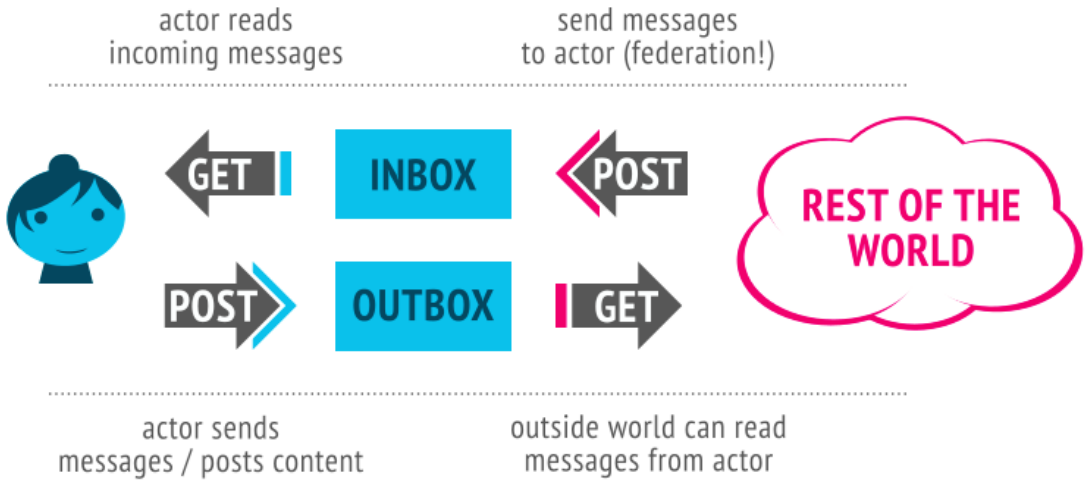
Referenz anpassen, sobald der Teil geschrieben wurde

Zusammengefasst muss sich der Client um das Entdecken von Endpunkten - Inbox/Outbox - über Aktorenobjekte, das Addressieren von Aktivitäten und Serialisieren von Daten zu Aktivitäten oder ActivityStreams 2.0 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.

Abbildung erstellen

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. Der Server Teil des Client-zu-Server Protokolls bezieht sich nur auf eine Domäne. Für das verteilen von Aktivitäten an andere Server muss zusätzlich das förderierte Server-zu-Server Protokoll implementiert werden.

2.3. Server-zu-Server Kommunikation



Quelle: ActivityPub 2018 - Overview

Abbildung 2.2.: Schnittstellen des ActivityPub Protokolls

Bildquelle richtig angeben..

Bei der Server-zu-Server Kommunikation bestehen die Hauptaufgaben im annehmen und zustellen von HTTP POST Anfragen auf die „Inboxen“ und „Outboxen“ der Nutzer.

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“ oder „Linked Data Signatures“zur Sicherstellung der Datenintegrität.

Datenintegrität

OAuth 2.0

HTTP Signaturen

Um sicherzustellen das HTTP Anfragen beim Transport nicht verändert wurden, können HTTP Signaturen verwendet werden. Bei diesen wird ein kryptografischer Algorithmus benutzt um

Linked Data Signaturen

Wenn ein Objekt nicht nur vom Client zum Server gesendet, sondern auch zwischen Servern untereinander weitergeleitet werden soll wird zum Sicherstellen der Datenintegrität ein anderes Verfahren benötig als HTTP Signaturen. Die „Best Practices“ empfehlen für solche Fälle „Linked Data Signatures“. Der größte Unterschied zwischen HTTP Signaturen und „Linked Data Signatures“ besteht darin, welche Daten zum Erstellen der Signatur verwendet werden. Bei HTTP Signaturen sind es die Kopfzeilen. Mit „Linked Data Signatures“ kann auch das Objekt selbst, also der Payload einer HTTP Anfrage, anstatt nur die Kopfzeilen, zum signieren verwendet werden.

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

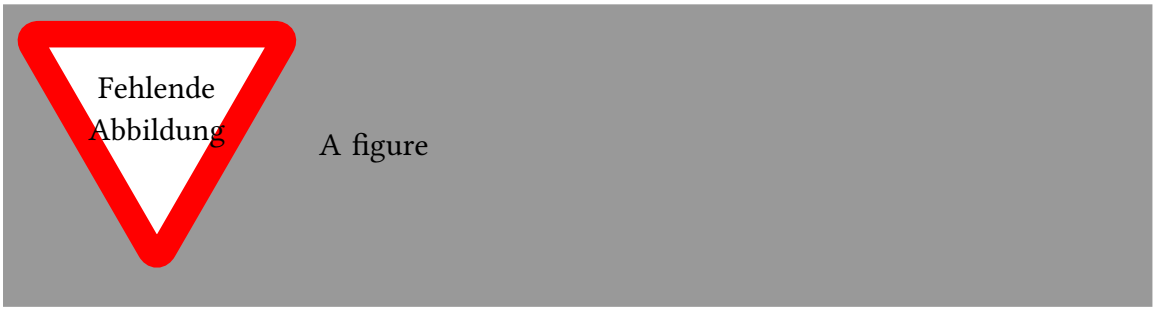


Abbildung A.1.: A figure

...