Fachhochschule Bielefeld  
Fachbereich Campus Minden

Studiengang M. Sc. Informatik

Evaluation von OpenID Connect und SAML in Hinblick auf Performance, UX und technische Möglichkeiten

Hausarbeit

Sommersemester 2022

Vorgelegt von: Rafael Berger

Matrikelnummer: 1181289

Abgabe am:

Prüfer: Prof. Dr. rer. nat. Jörg Brunsmann

**Inhaltsverzeichnis**

[1 Einleitung - 1 -](#_Toc103782971)

[1.1 Motivation - 1 -](#_Toc103782972)

[1.2 Ziel - 1 -](#_Toc103782973)

[1.3 Vorgehensweise - 1 -](#_Toc103782974)

[2 Unterschied Autorisierung und Authentifizierung - 2 -](#_Toc103782975)

[3 OAuth 2.0 - 2 -](#_Toc103782976)

[3.1 Tokens - 2 -](#_Toc103782977)

[3.2 Rollen - 2 -](#_Toc103782978)

[3.3 Autorisierungsabläufe - 3 -](#_Toc103782979)

[3.3.1 Authorization Code Grant - 3 -](#_Toc103782980)

[3.3.2 Implicit Grant - 4 -](#_Toc103782981)

[3.3.3 Resource Owner Password Credentials Grant - 5 -](#_Toc103782982)

[3.3.4 Client Credentials Grant - 5 -](#_Toc103782983)

[4 OpenID Connect - 6 -](#_Toc103782984)

[4.1 JWT - 6 -](#_Toc103782985)

[4.1.1 Header - 6 -](#_Toc103782986)

[4.1.2 Payload - 6 -](#_Toc103782987)

[4.1.3 Signatur - 7 -](#_Toc103782988)

[4.2 ID-Token - 7 -](#_Toc103782989)

[5 SAML - 8 -](#_Toc103782990)

[5.1 Begriffe - 8 -](#_Toc103782991)

[5.2 Authentifizierungsablauf - 9 -](#_Toc103782992)

[6 Unterschiede zwischen SAML und OIDC - 10 -](#_Toc103782993)

[7 Implementierung - 11 -](#_Toc103782994)

[8 Zusammenfassung - 12 -](#_Toc103782995)

[8.1 Zusammenfassung - 12 -](#_Toc103782996)

[9 Literaturverzeichnis - 13 -](#_Toc103782997)

[10 Anhang - 14 -](#_Toc103782998)

Abbildungsverzeichnis

[Abbildung 1: Authorization Code Grant - 3 -](#_Toc103782999)

[Abbildung 2: Implicit Grant - 4 -](#_Toc103783000)

[Abbildung 3: Resource Owner Password Credentials Grant - 5 -](#_Toc103783001)

[Abbildung 4: Client Credentials Grant - 5 -](#_Toc103783002)

[Abbildung 5: SAML-Deep-Links - 9 -](#_Toc103783003)

Tabellenverzeichnis

**Es konnten keine Einträge für ein Abbildungsverzeichnis gefunden werden.**

Abkürzungsverzeichnis

|  |  |
| --- | --- |
| Abkürzung | Beschreibung |
|  |  |

1. Einleitung
   1. Motivation
   2. Ziel
   3. Vorgehensweise
2. Unterschied Autorisierung und Authentifizierung

* Autorisierung
  + Schutz vor Ressource
  + Erlaubnis wird benötigt, um auf eine bestimmte Ressource zuzugreifen
  + Beispiel
    - Parkhilfe
* Authentifizierung
  + Stellt sicher, dass die angegebene Person oder der Service auch korrekt ist
  + Handelt es sich wirklich um die Person oder der Service für den er sich ausgibt
  + Beispiel
    - Hotelkarte an der Rezeption

1. OAuth 2.0

* Standard der 2012 entwickelt wurde
* Dient zur Autorisierung mittels eines Access-Tokens
  1. Tokens
* Access-Token
  + Notwendig für die Autorisierung
* Refresh-Token
  + Aktualisiert den Access-Token, damit dieser nicht abläuft
* Code-Token
  + Wird in speziellen Autorisierungsabläufe gebraucht, um initial ein Access-Token und Refresh-Token zu erhalten
  1. Rollen
  + Ressourcen-Besitzer
  + Client
  + Autorisierungsserver
  + Ressourcen-Server
  1. Autorisierungsabläufe
     1. Authorization Code Grant

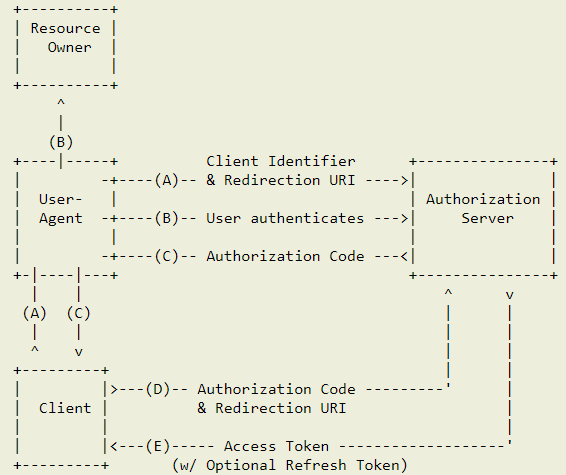


Abbildung 1: Authorization Code Grant

* + 1. Implicit Grant

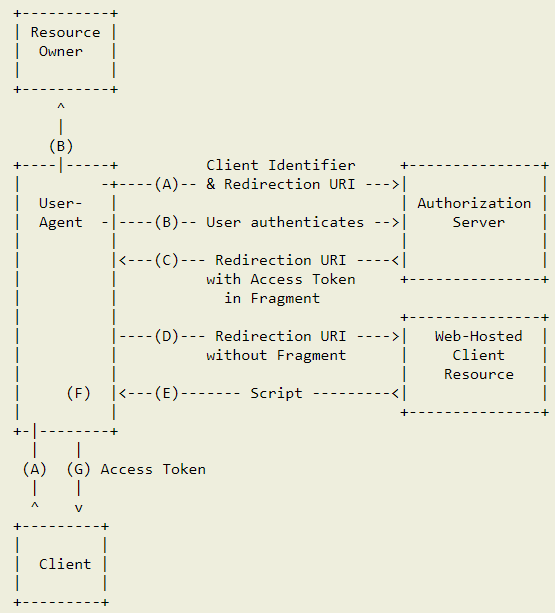


Abbildung 2: Implicit Grant

* + 1. Resource Owner Password Credentials Grant

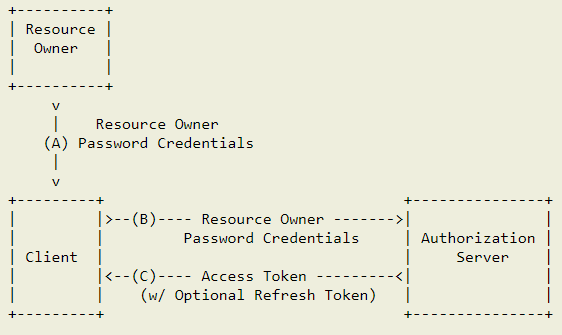


Abbildung 3: Resource Owner Password Credentials Grant

* + 1. Client Credentials Grant

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

Abbildung 4: Client Credentials Grant

1. OpenID Connect
   1. JWT

* Offener Standard (RFC 7519)
* Dient zur sicheren Übermittlung von Daten als JSON-Objekt
* Durch eine digitale Signierung können die Daten als vertrauenswürdig angesehen werden
  + Informationen sind nicht manipuliert worden
* Werden häufig bei der Autorisierung und Informationsaustausch genutzt
* Single Sign On ist eine beispielhafte Funktion für die JWT verwendet werden
* Ein JWT besteht aus drei Komponente die jeweils mit einem Punkt getrennt werden
  + xxx.yyy.zzz
    1. Header
* Besteht aus zwei Teilen
  + Typ des Tokens
  + Signaturalgorithmus
    1. Payload
* Enthält alle Informationen die gesendet werden
  + Können Daten über den Benutzer enthalten
* Die enthaltenen Behauptungen werden Claims genannt
* Es gibt drei verschiedene Typen von Claims
  + Registrierte
    - Vordefinierte Behauptungen die nicht erforderlich sind aber empfohlen werden
      * iss (Aussteller)
      * exp (Ablaufzeit)
      * sub (Betreff)
      * aud (Zielgruppe)
  + Öffentliche
    - Können beliebig definiert werden
    - Sollte aber in der IANA JSON Web Token Registry enthalten sein und sich daran orientieren, um Kollisionen zu vermeiden
  + Private
    - Können frei definiert werden
    - Sollten aber nur in der eigenen Umgebung eingesetzt werden
    1. Signatur
* Header und Payload werden verwendet mit zusätzlich einem Geheimnis, um die Signatur zu erstellen
  1. ID-Token
* Besonderheit im Gegensatz zu OAuth2.0
* JWT, dass als Sicherheitstoken für die Authentifizierung funktioniert
* Es sind einige Claims definiert, die dieses ID-Token enthalten muss
  + iss
  + sub
  + aud
  + exp
  + iat
* Andere sind optional
  + acr
  + amr
  + azp
* Zwei weitere können erforderlich oder optional sein. Kommt dabei auf die Implementierung an
  + auth\_time
  + nonce

1. SAML

* Für die Authentifizierung wird eine „Assertion“ verwendet
* Die Assertion ist ein XML-Dokument, welches verschiedene Informationen enthält
  + Wer hat sich authentifiziert
  + Informationen über die Person
  + Wer die Assertion erstellt hat
  + Weitere Daten für die Authentifizierung
* Sendet die Assertion meistens direkt über den Browser
  + Verwenden die POST-Bindung für das Senden der Antwort
  + Der Browser sendet dabei ein HTML-Formular, welches vom Identitätsanbieter erstellt wird, mit der XML für die Antwort als Formularparameter
    - Für das Versenden der Daten an den Dienstanbieter, enthält das Formular JavaScript
    - Dadurch besteht keine direkte Kommunikation zwischen dem Dienstanbieter und dem Identitätsanbieter, da die Informationen immer über den Browser weitergeleitet werden
  1. Begriffe
* Service Provider (SP)
  + Häufig die Anwendung
* Identity Provider (IdP)
  + Entität die für die Authentifizierung zuständig ist
* SAML Request
  + Wird vom Service Provider generiert, damit die Authentifizierung angefordert wird
* SAML Response
  + Die Antwort vom Identity Provider
  + Enthält die Assertion vom Benutzer
  + Kann zusätzliche Informationen enthalten
    - Benutzerprofil
    - Gruppen-/Rolleninformationen
* Service Provider Initiated (SP-initiated)
  + Beschreibt den Ablauf der SAML-Anmeldung
  + Wird ausgelöst, falls versucht wird direkt auf eine geschützte Ressource zuzugreifen
  + Wird ausgelöst vom Service Provider (Dienstleister)
* Identity Provider Initiated (IdP-initiated)
  + Beschreibt den Anmeldeablauf, der vom Identity Provider ausgelöst wird
  1. Authentifizierungsablauf

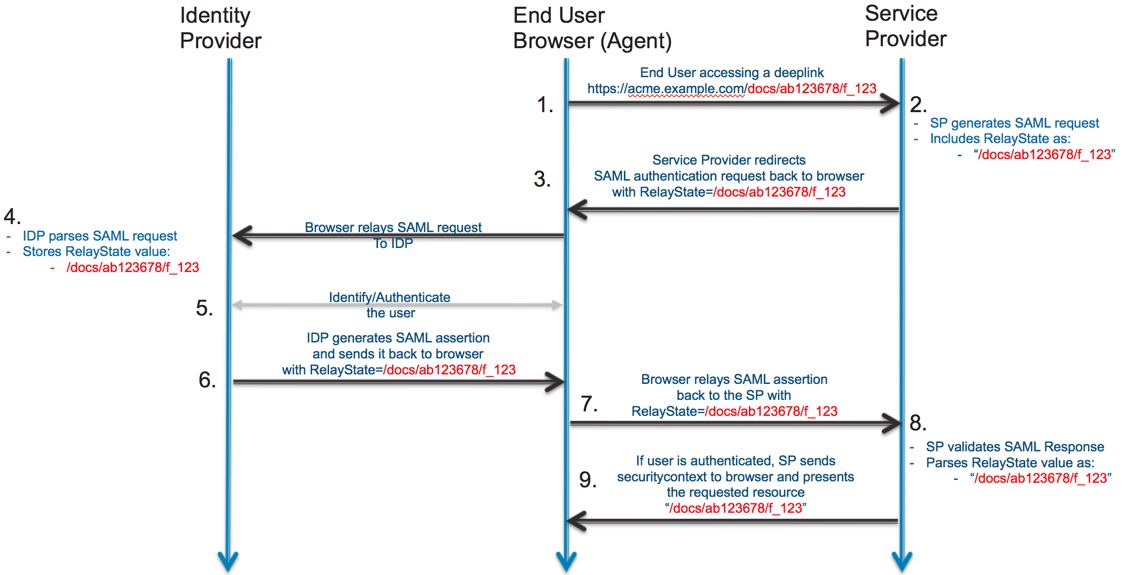


Abbildung 5: SAML-Deep-Links

1. Unterschiede zwischen SAML und OIDC

|  |  |  |
| --- | --- | --- |
|  | OIDC | SAML |
| Tokenart | XML | JWT |
| Kommunikation Anwendung und IDP/OP | Direkt | Über den Browser |
| Mobile Anwendung | Ja | Nein |
| Anwendung unterstützt bereits SAML | Nein | Ja |
| Neue Anwendung | Ja | Nein |

1. Implementierung
2. Zusammenfassung
   1. Zusammenfassung
3. Literaturverzeichnis
4. Anhang

Anhangsverzeichnis

Es konnten keine Einträge für ein Abbildungsverzeichnis gefunden werden.

Eidesstattliche Erklärung

Ich erkläre hiermit an Eides Statt, dass ich die vorliegende Hausarbeit selbstständig angefertigt habe. Die aus fremden Quellen direkt oder indirekt übernommenen Gedanken sind als solche kenntlich gemacht.

Die Arbeit wurde bisher weder in gleicher noch in ähnlicher Form einer anderen Prüfungsbehörde vorgelegt und auch noch nicht veröffentlicht.

Datum, Unterschrift