B2B- und B2C-Szenarien mit Azure

Abstract

Aus Digitalisierung und Zusammenarbeit ergibt sich auch immer die Fragestellung, wie mit Logins und Berechtigungen umgegangen wird. Das Microsoft-Toolset, wie z. B. Teams, bedient sich hier aus allgemeingültigen Konzepten und Protokollen, welche mit Azure umgesetzt sind.
Im Vortrag wird auf die Unterschiede von B2B- und B2C-Szenarien eingegangen und eine technische Einführung in Azure AD gegeben.

Vorstellung

Melanie Eibl

mail@melanie-eibl.de

@melanie9701

Freiberufliche Beraterin

dotnet Cologne

Meetup Azure Bonn

Modern Authentication

Was ist das eigentlich?

Azure AD

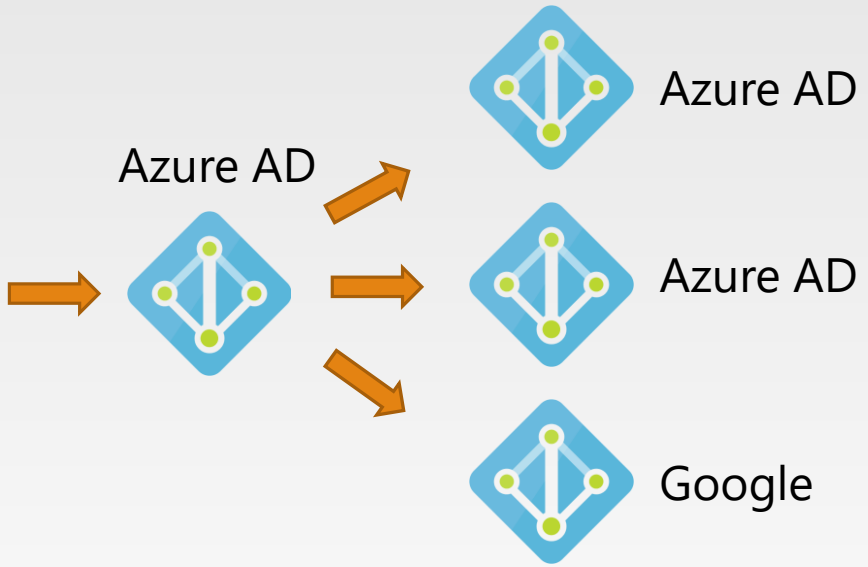
- Identity Provider = IdP = IP = Authority
- Root-Element eines Azure Tenants
- Enthält
  - Directory-Roles
  - Gruppen

Demo: Azure AD

Was ist Azure AD B2B? (1)

Funktion, mit der man einen Benutzer aus einem anderen AD hinzufügen kann und zur Bearbeitung der selben Anwendungen und Ressourcen einladen kann

Keine weitere Verzeichnisart



Was ist Azure AD B2B? (2)

Was ist Azure AD B2B?

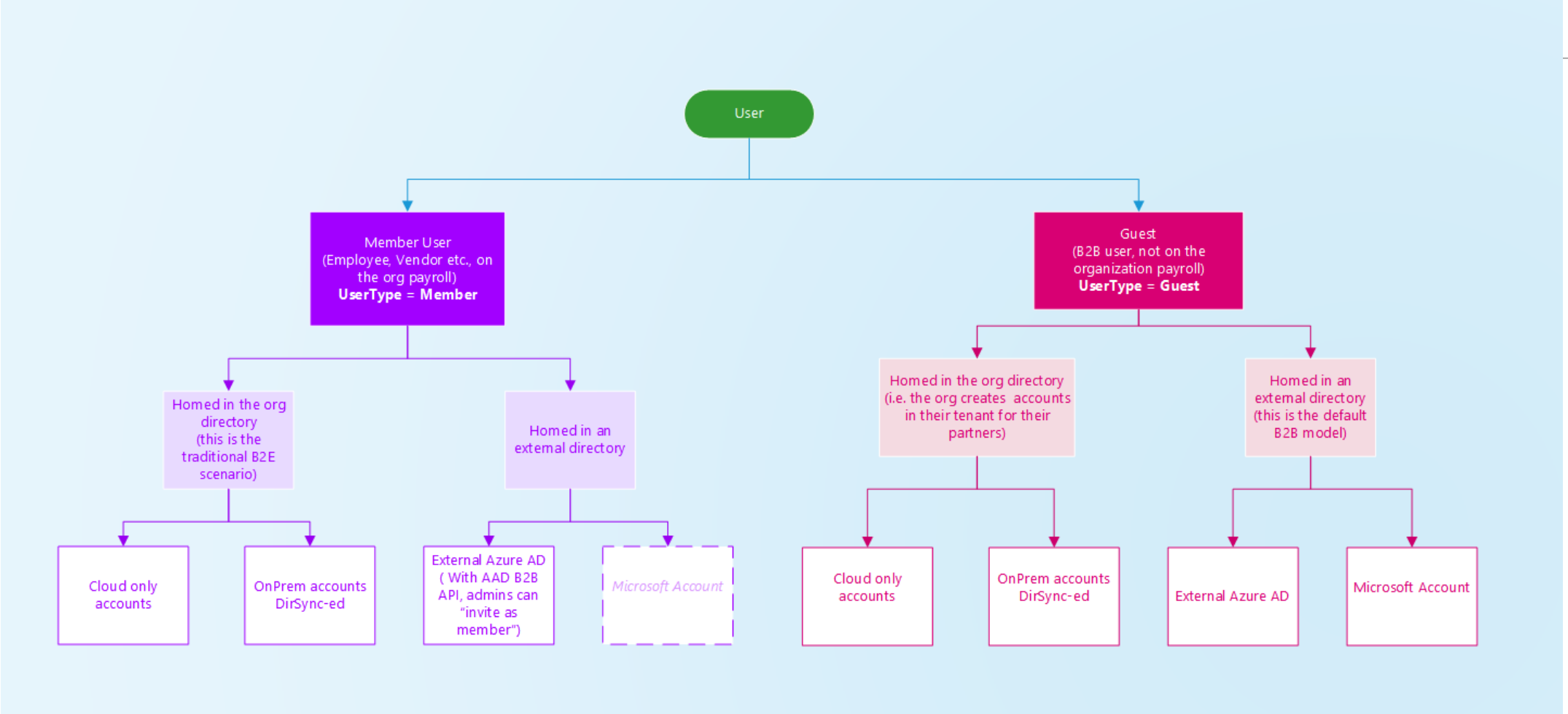
Vorteile:

- Wenn Benutzer aus einer anderen Organisationen eingeladen werden, werden sie als Objekte im Verzeichnis erstellt. Die gesamte Kontoverwaltung bleibt jedoch beim ursprünglichen Tenant.
- Sie können genau steuern, auf welche Ressourcen dieses Konto in Ihrem Mandanten Zugriff hat.
- Möglichkeit, eigene Policies (Richtlinien) auf den Gast anzuwenden. Wenn MFA für den Zugriff auf Ressourcen erforderlich ist und dies nicht im ursprünglichen Tenant konfiguriert ist, kann das hier nachgeholt werden.
- Möglichkeit der Zugriffssteuerung über Conditional Access

Externe User ohne Beeinträchtigung der Sicherheitsrichlinien der eigenen Organisation

Consumer Grade Account => Microsoft Account

Eigenschaften eines Collab-Users



https://docs.microsoft.com/en-us/azure/active-directory/b2b/user-properties

Teams

Teams setzt auf Office 365 auf und Office 365 auf Azure, inkl. Azure AD

Zusammenarbeit mit Benutzern außerhalb des eigenen Unternehmens => Guest-Access war das am meisten gefragte Feature für Teams

Single-Singn-On über die Nutzung von Active Directory

Teambesitzer kann die Mitglieder verwalten

Jeder Benutzer mit einem „Business“ oder „Consumer“ Account kann einem Team hinzugefügt werden

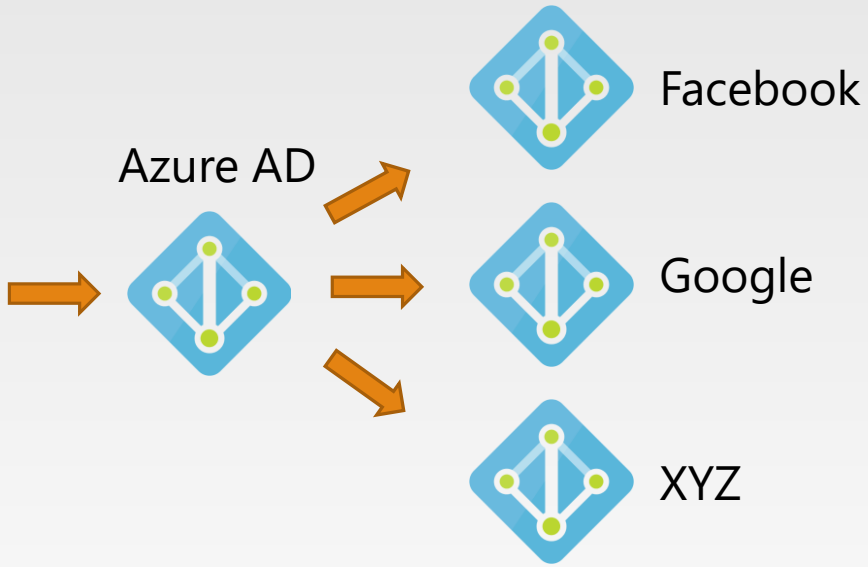
Demo: Gastbenutzer in Teams

Was ist Azure AD B2C?

Sonderform des Azure AD

User
- Local Account
- Federated Account(s)

security token service (STS)
- ID tokens
- refresh tokens
- access tokens
- and equivalent SAML assertions



Besonderheiten Azure AD B2C

Erweiterbares Policy-Framework => Ermöglicht die Beschreibung der vollständigen Identity-Experience

- Sign-Up
- Sign-In
- Password-Change
- Profile-Edit

=> User-Self-Service

MFA

SSO

Anbindung weiterer Kontotypen (Facebook, Google, beliebiger Identity Provider, der OpenID Connect unterstützt)

Schützt vor Denial-of-Service- und Kennwortangriffen

Verwendet Erkennungs- und Schadensbegrenzungstechniken wie SYN-Cookies sowie Geschwindigkeits- und Verbindungslimits, um Ressourcen vor Denial-of-Service-Angriffen zu schützen

Mitigation ist auch für Brute-Force-Passwort-Angriffe und Dictionary-Passwort-Angriffe enthalten.

Demo: Anlegen B2C-Tenant

Identity Experience Framework – Custom Policies

=> Public Preview

The underlying platform that establishes multi-party trust

This framework and a policy (also called a user journey or a Trust Framework policy) explicitly defines the actors, the actions, the protocols, and the sequence of steps to complete.

Custom Policies

- XML-Files
  - Base file
  - Extensions file
  - Relying Party (RP) file

Ändern des Benutzernamens

Reines Sign-In mit gleichzeitiger UI-customization

Besondere Anforderungen zur User Journey (API-Calls)

Single-Sign-On

SSO kann innerhalb der Policies konfiguriert werden

Scope kann sein:

- Tenant (default)
- Application
- Policy
- Disabled

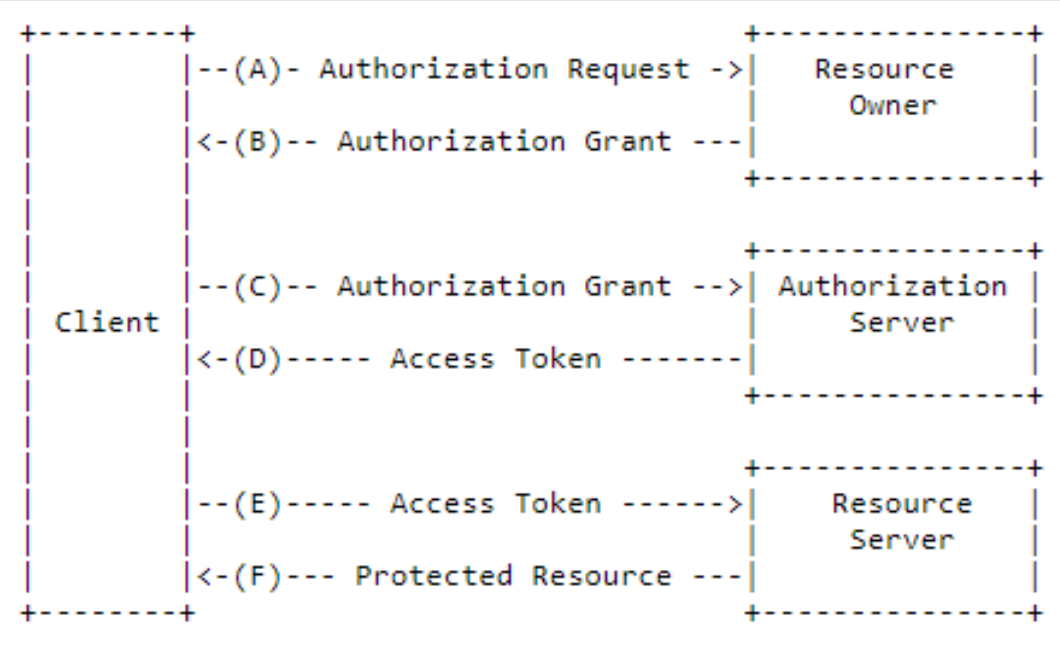
MFA kann konfiguriert werden

OAuth 2.0 – Abstrakter Protokollfluss

Resource Owner: Entität, die Zugriff auf eine Ressource gewähren kann

Resource Server: Server, der eine Ressource hostet

Client: Applikation, die auf eine Ressource im Auftrag des Resource Owners zugreifen möchte



Authorization Code Grant

Für vertrauenswürdige Clients, da authorization code herausgegeben wird

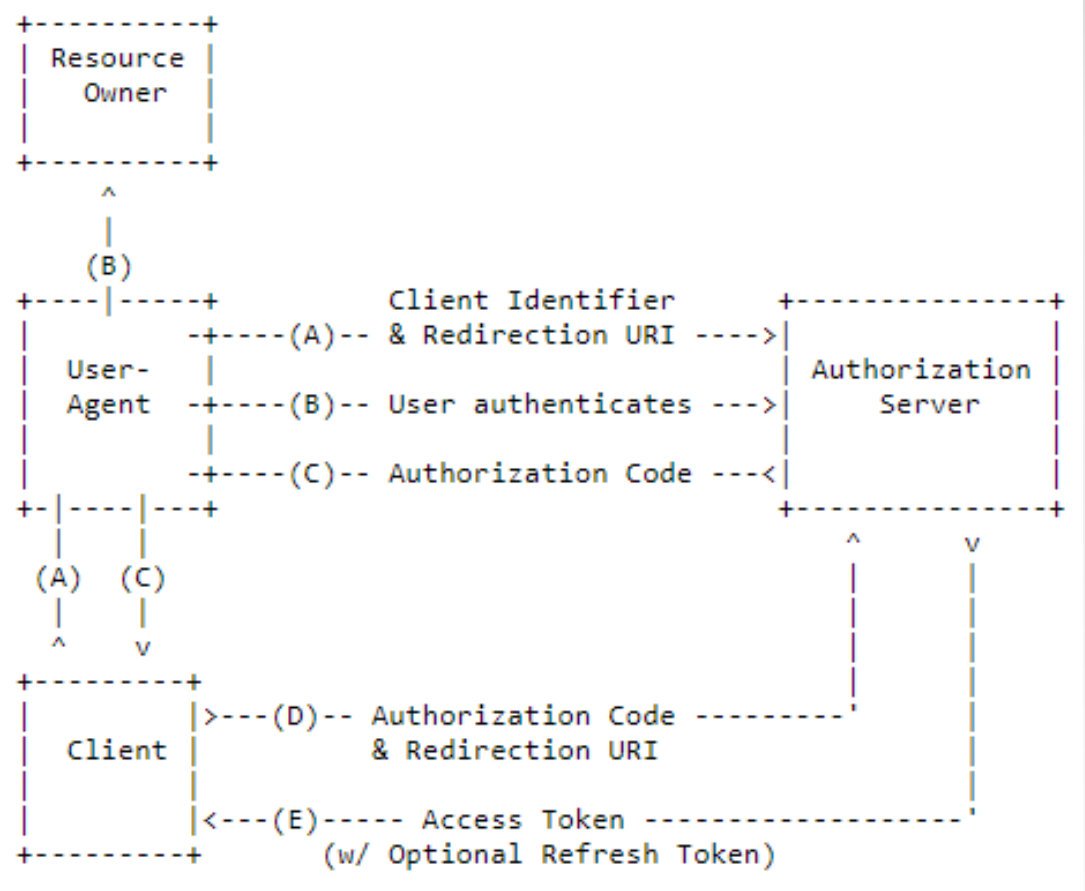
Access- und (optional) Refresh-Token

Basiert auf Redirect

Client muss sich authentifizieren

Redirect URI muss passen

Separate Requests für Authorization und Access Token



Implicit Grant

Für nicht vertrauenswürdige Clients (SPA)

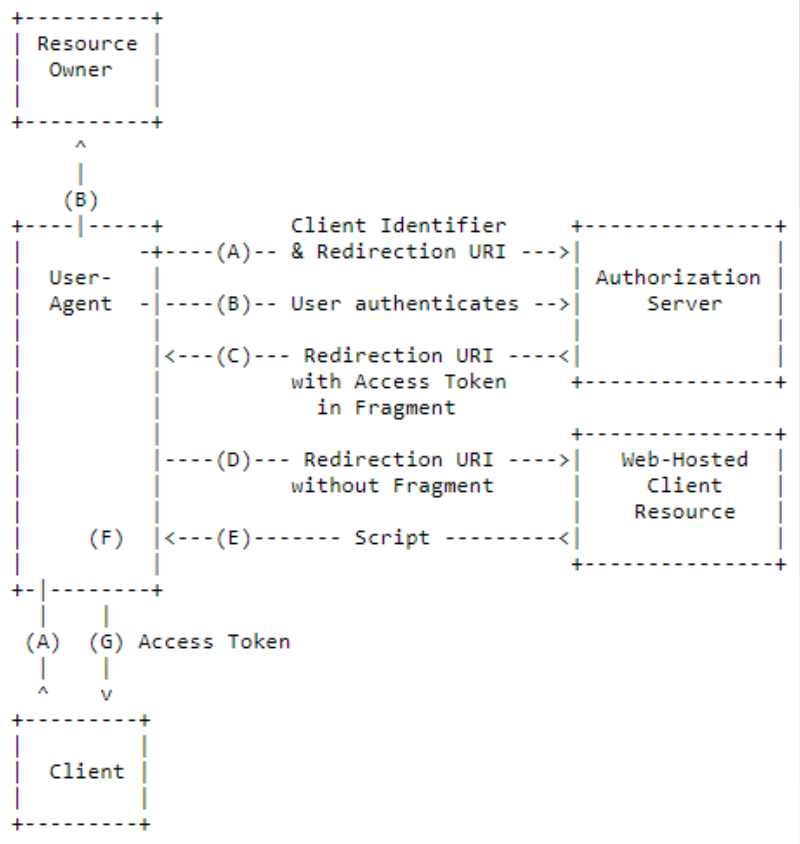
Nur Access Token in Redirect URI

Basiert auf Redirect

Anders als beim Authorization Code Grand Type,

Keine Client Authentifizierung

Request für Authorization und Access Token zusammengefasst

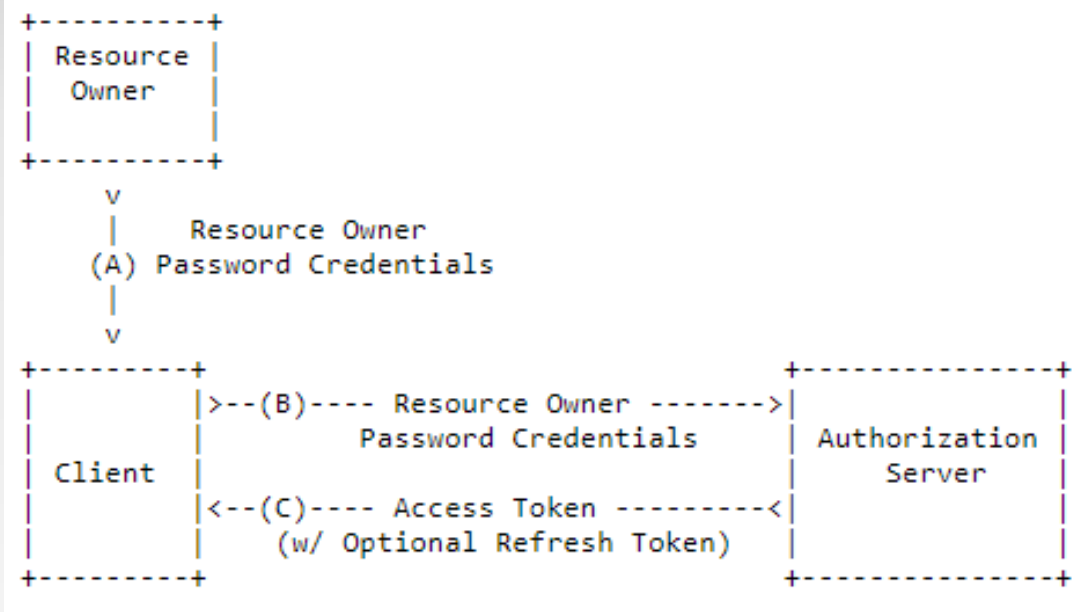


Resource Owner Password Credentials Grant

4.3. Resource Owner Password Credentials Grant

The resource owner password credentials grant type is suitable in cases where the resource owner has a trust relationship with the client, such as the device operating system or a highly privileged application. The authorization server should take special care when enabling this grant type and only allow it when other flows are not viable.

This grant type is suitable for clients capable of obtaining the resource owner's credentials (username and password, typically using an interactive form). It is also used to migrate existing clients using direct authentication schemes such as HTTP Basic or Digest authentication to OAuth by converting the stored credentials to an access token.

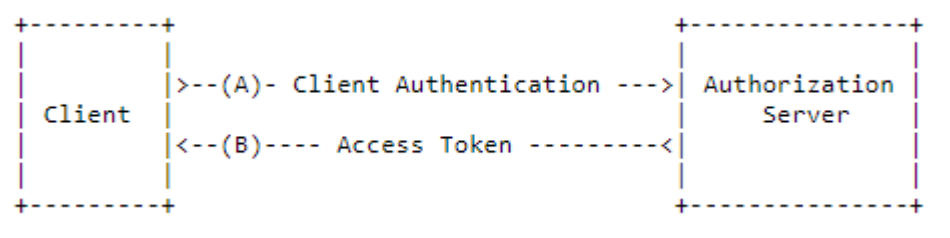


Client Credentials Grant

Reine Client Authentifizierung

Access Token wird ohne Interaktion mit einem User herausgegeben

Technische Identität



OpenID Connect

Es ist einfach, zuverlässig und sicher

Funktioniert mit Web-, Desktop- und mobilen Applikationen

Gute Unterstützung durch Bibliotheken

Verantwortlichkeit für den Schutz der Identität liegt beim Identity Provider

Zwei sog. „Implementer‘s Guide“

- OpenID Connect Basic Client Implementer's Guide => setzt auf dem OAuth Authorization Code Flow auf
- OpenID Connect Implicit Client Implementer's Guide => setzt auf dem OAuth Implicit Flow auf

Informationen über den Benutzer in „Claims“

In welcher Beziehung steht OpenID Connect zu SAML?

Security Assertion Markup Language (SAML) ist XML-basiert und eine Federation-Technologie

OpenID Connect kann dieselben Anwendungsfälle erfüllen, jedoch mit einem einfacheren, auf JSON / REST basierenden Protokoll

OpenID Connect wurde entwickelt, um auch native Apps und mobile Anwendungen zu unterstützen, während SAML nur für webbasierte Anwendungen entwickelt wurde.

SAML und OpenID Connect werden wahrscheinlich noch einige Zeit koexistieren.

Fragen?

Danke!