

Governing the "Shadow AI" Mesh with Open Standards

Aaron Parecki

Director of Identity Standards

Okta

Updated by: [8252](#)

PROPOSED STANDARD

[Errata Exist](#)

Internet Engineering Task Force (IETF)

D. Hardt, Ed.

Request for Comments: 6749

Microsoft

Obsoletes: [5849](#)

October 2012

Category: Standards Track

ISSN: 2070-1721

The OAuth 2.0 Authorization Framework

Abstract

The OAuth 2.0 authorization framework enables a third-party application to obtain limited access to an HTTP service, either on behalf of a resource owner by orchestrating an approval interaction between the resource owner and the HTTP service, or by allowing the third-party application to obtain access on its own behalf. This specification replaces and obsoletes the OAuth 1.0 protocol described in [RFC 5849](#).

Status of This Memo

This is an Internet Standards Track document.

This document is a product of the Internet Engineering Task Force (IETF). It represents the consensus of the IETF community. It has received public review and has been approved for publication by the Internet Engineering Steering Group (IESG). Further information on Internet Standards is available in [Section 2 of RFC 5740](#).

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at <http://www.rfc-editor.org/info/rfc6749>.

Copyright Notice

Copyright (c) 2012 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to [BCP 78](#) and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

- [1. Introduction](#)4
- [1.1. Roles](#)6
- [1.2. Protocol Flow](#)7
- [1.3. Authorization Grant](#)8
 - [1.3.1. Authorization Code](#)8
 - [1.3.2. Implicit](#)8
 - [1.3.3. Resource Owner Password Credentials](#)9
 - [1.3.4. Client Credentials](#)9
- [1.4. Access Token](#)10
- [1.5. Refresh Token](#)10
- [1.6. TLS Version](#)12
- [1.7. HTTP Redirections](#)12

Specs are not good tutorials!



 **Secure** | <https://yelp.com/>

 **Sign in with Facebook**

 **Sign in with Google**

 **Sign in with LinkedIn**

 **Sign in with Twitter**

The Password Anti-Pattern



Real People. Real Reviews.™

Search for (e.g. taco, salon, Max's)

Near (Address, Neighborhood, City, State or Zip)

[Welcome](#) [About Me](#) [Write a Review](#) [Find Reviews](#) [Invite Friends](#) [Messaging](#) [Talk](#) [Events](#) [Member](#)

Are your friends already on Yelp?

Many of your friends may already be here, now you can find out. Just log in and we'll display all your contacts, and you can select which ones to invite. We don't keep your email password or your friends' addresses. We loathe spam, too.

Your Email Service

 Hotmail  YAHOO! MAIL  AOL Mail 

Your Email Address

(e.g. bob@yahoo.com)

Your Yahoo Password

(The password you use to log into your Yahoo email)



The Password Anti-Pattern

Step 1
Find Friends

Step 2
Profile Information

Step 3
Profile Picture

Are your friends already on Facebook?
Many of your friends may already be here. Searching your email account is the fastest way to find your friends on Facebook.

 **Gmail**

Your Email:

Email Password:

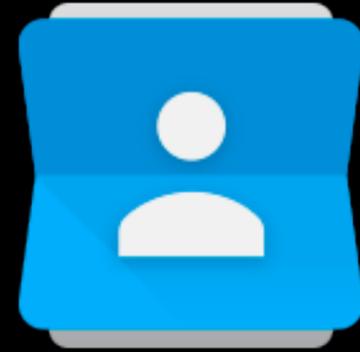
[Find Friends](#)

 Facebook will not store your password.

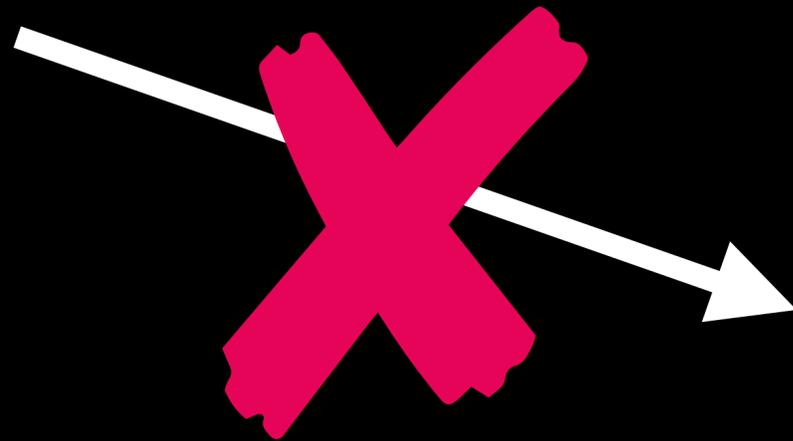
 **Yahoo!** [Find Friends](#)

 **Windows Live Hotmail** [Find Friends](#)

 **Other Email Service** [Find Friends](#)



Google Contacts



how can I let an app

access my data

without giving it my password?



Authorization Server



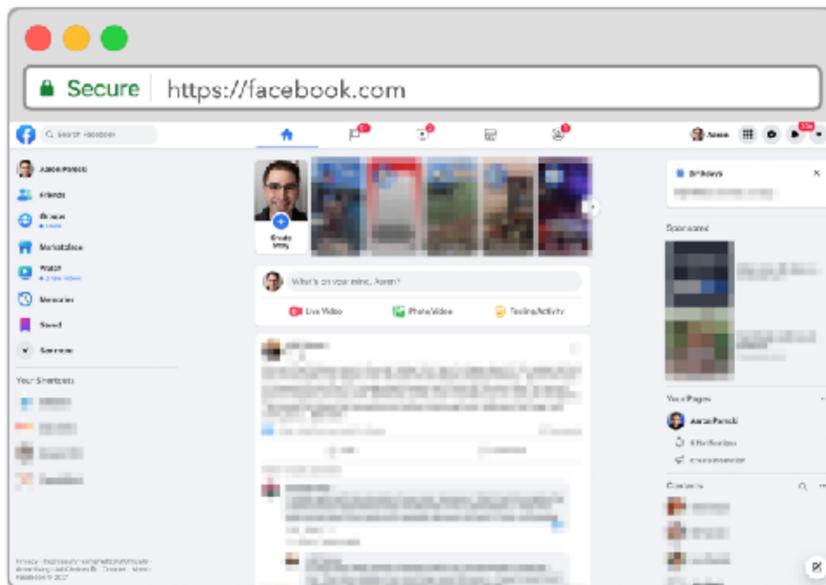
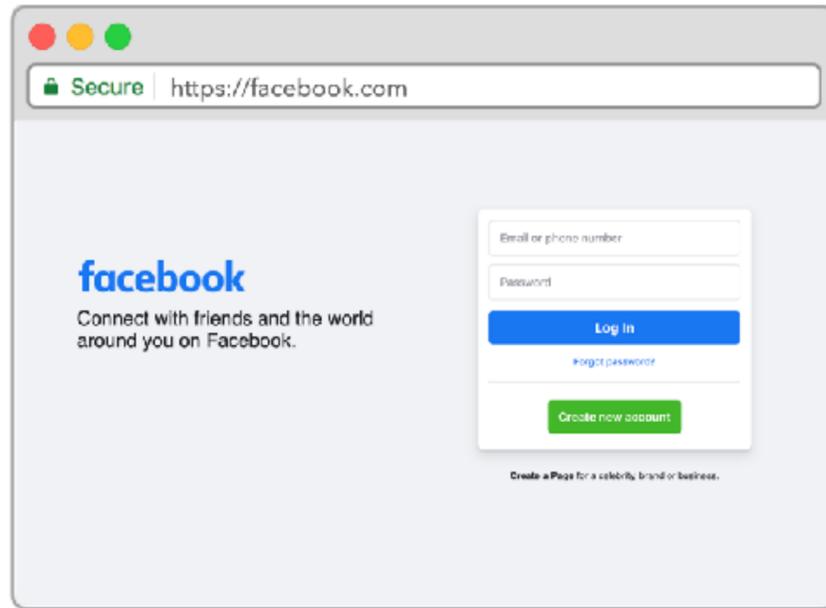
Access Token



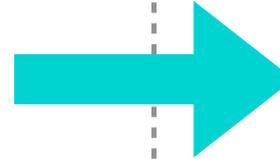
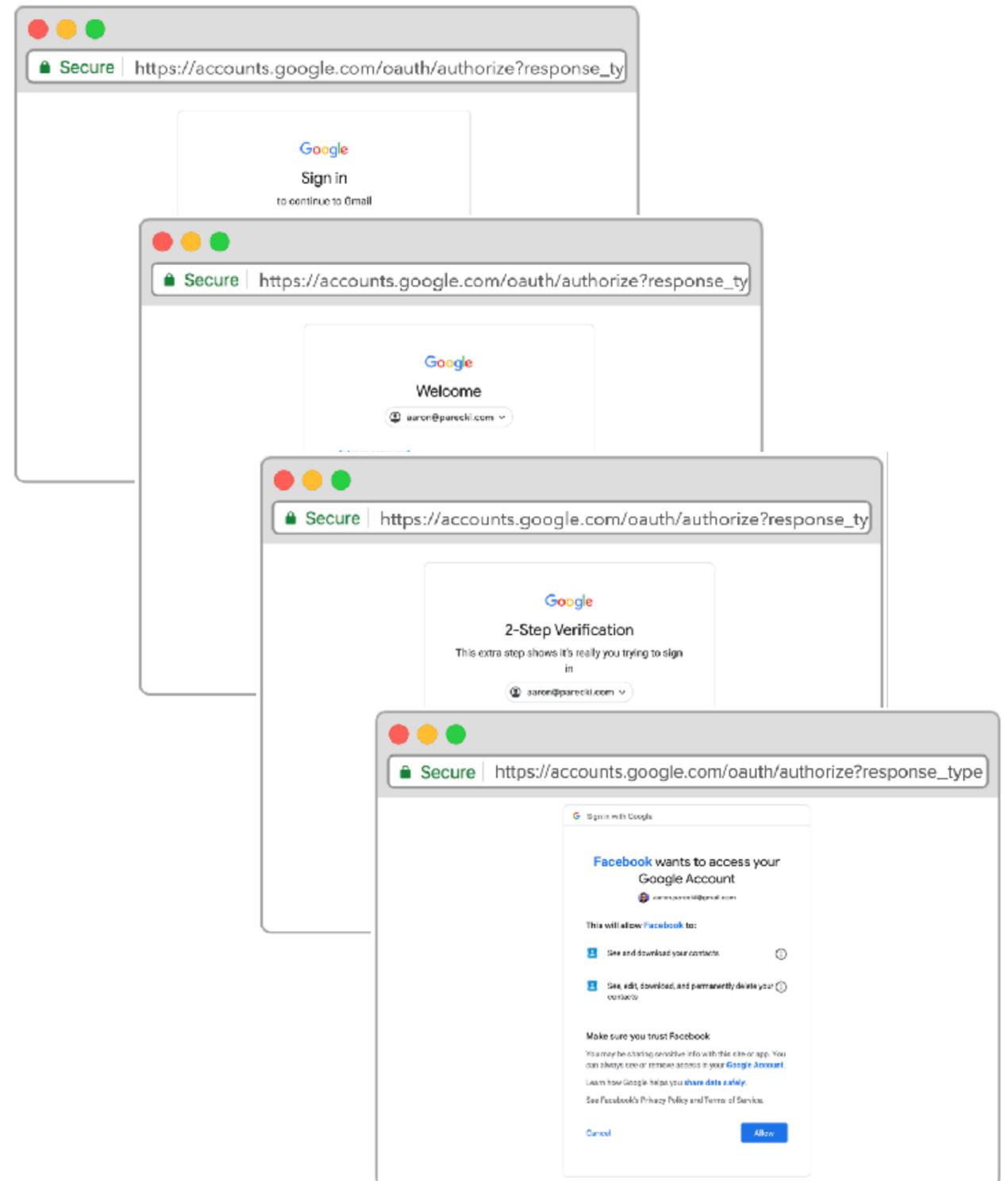
Resource (API)

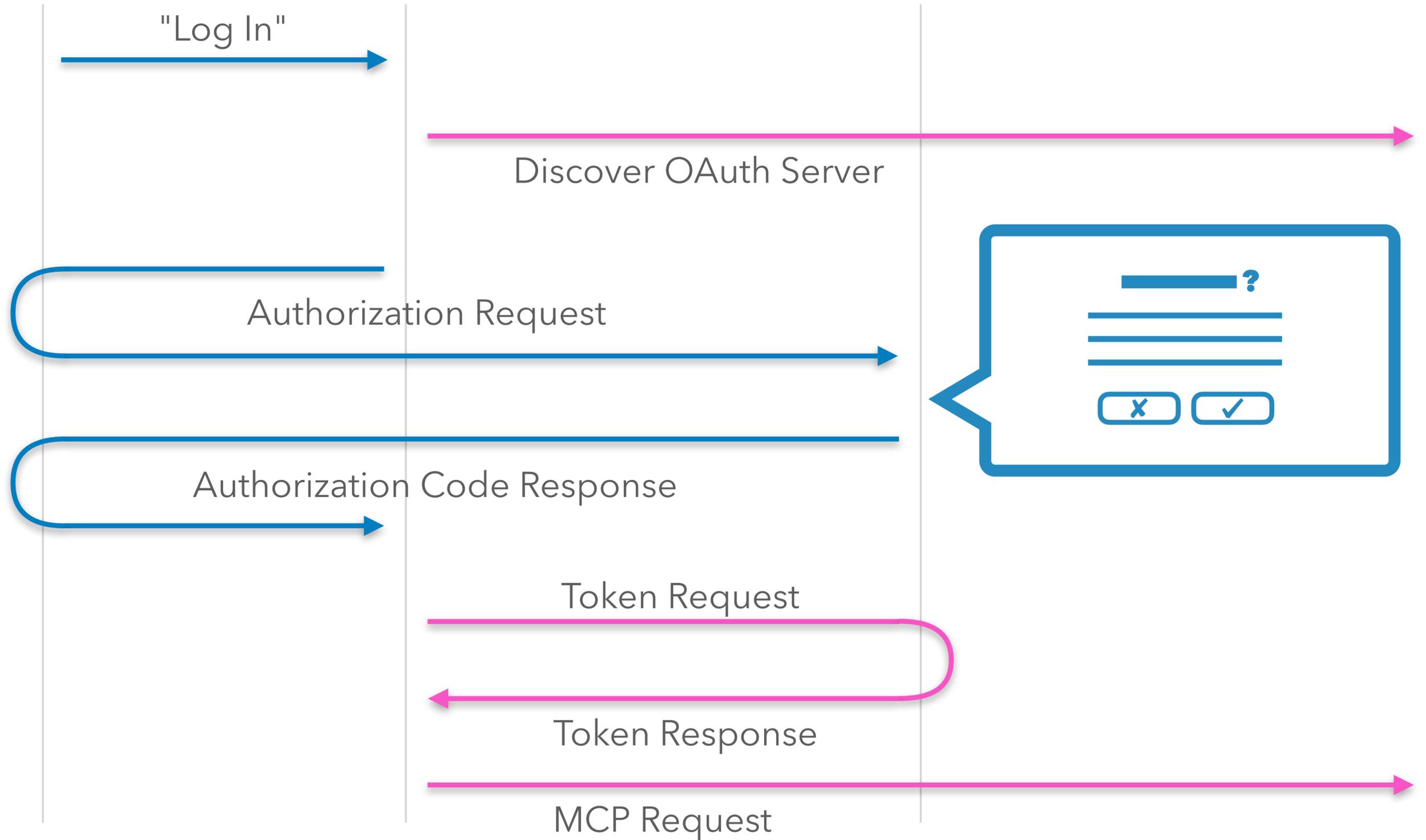


Application



OAuth Server







Gives the application a way to make API requests

Accessing APIs



Tells the application about the user authenticating

Identification



Gives the application a way to make API requests

Access Token



Tells the application about the user authenticating

ID Token



Access Token

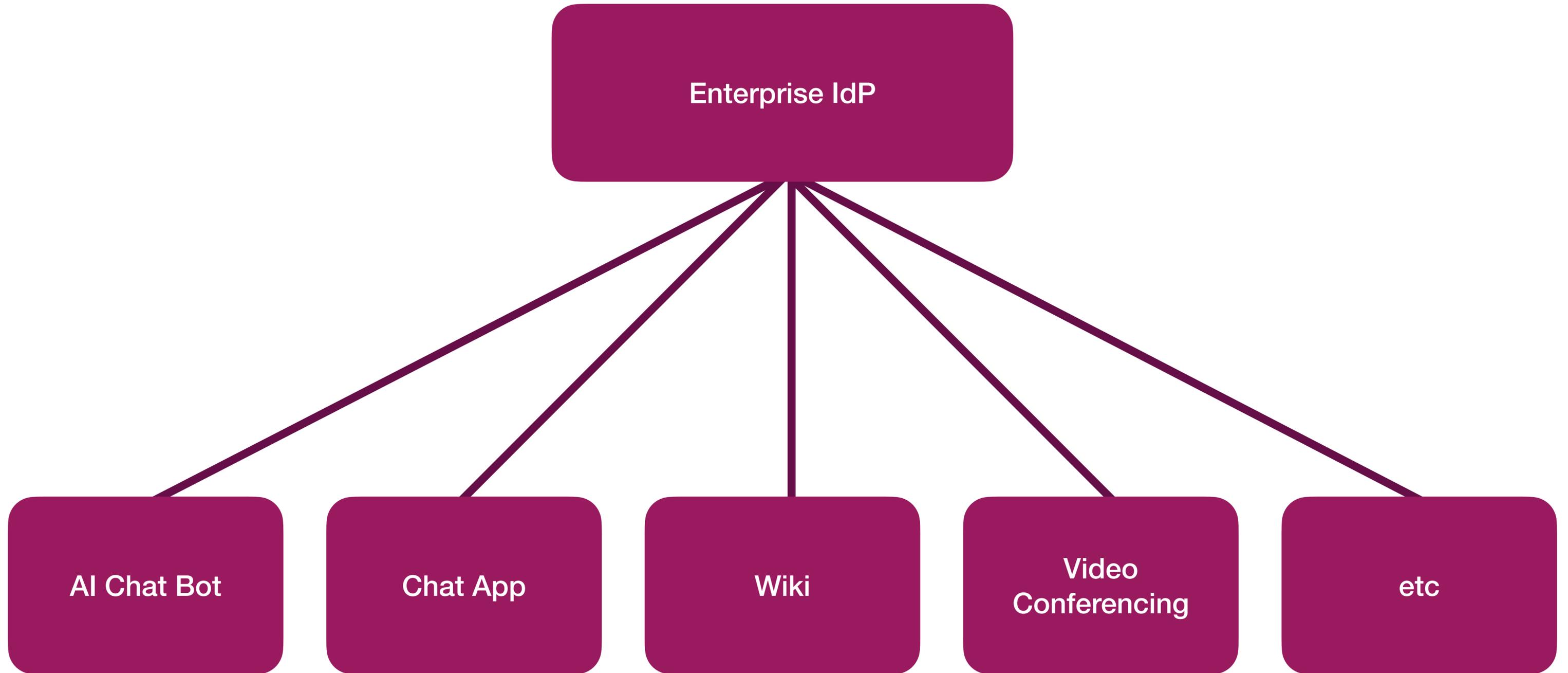


ID Token

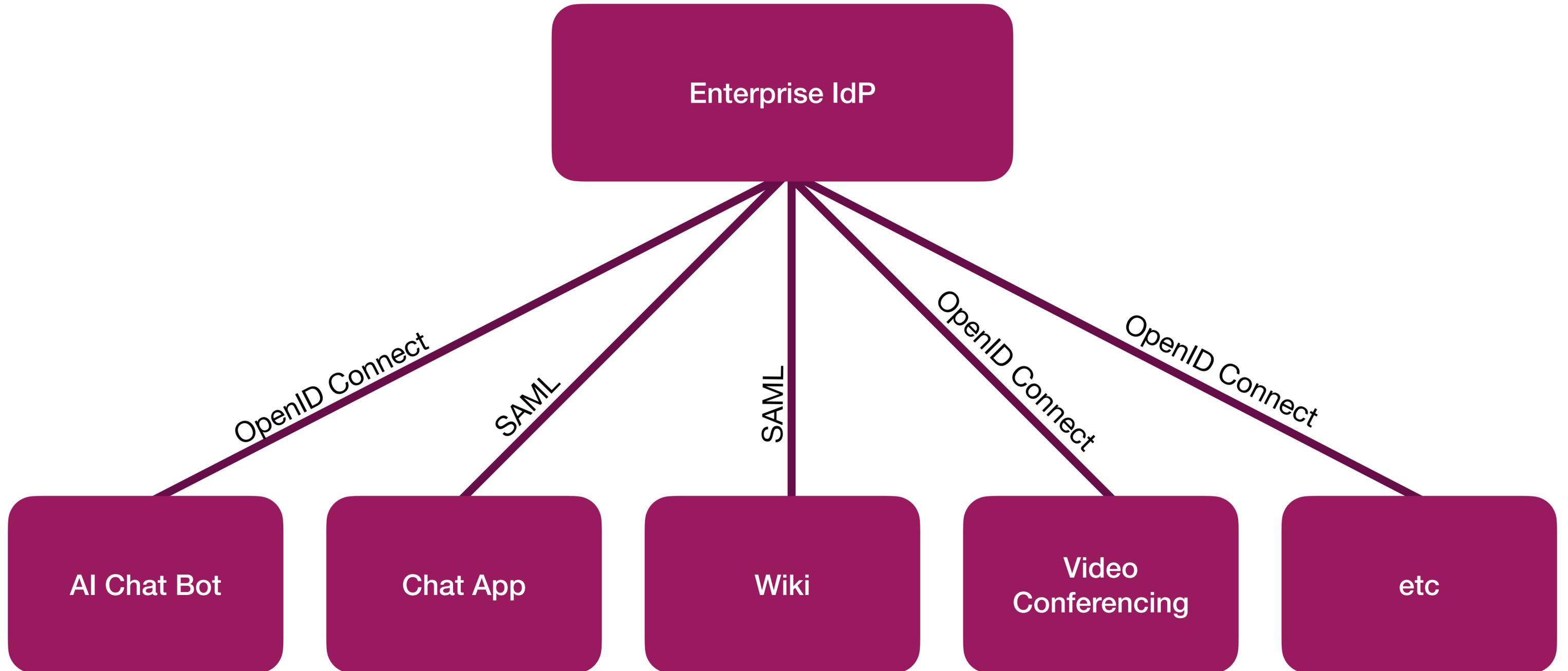


Enterprise IdP

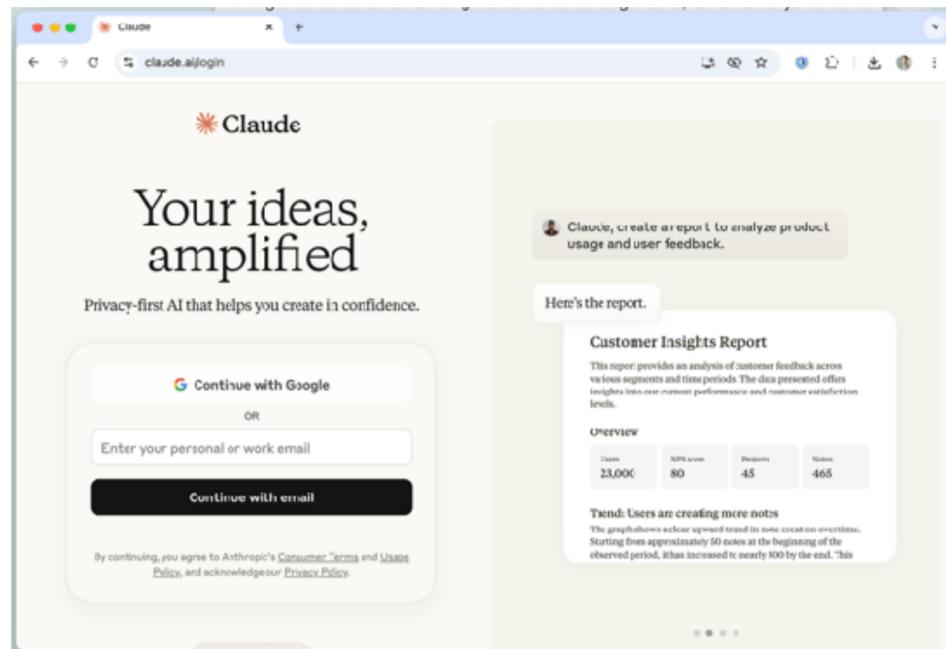
Enterprise Single Sign-On



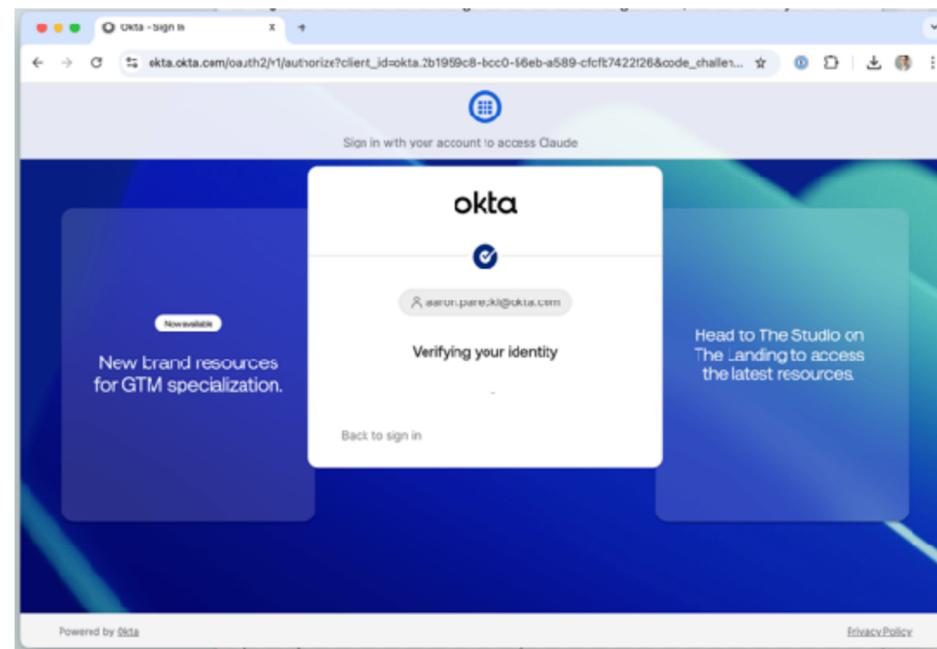
Enterprise Single Sign-On



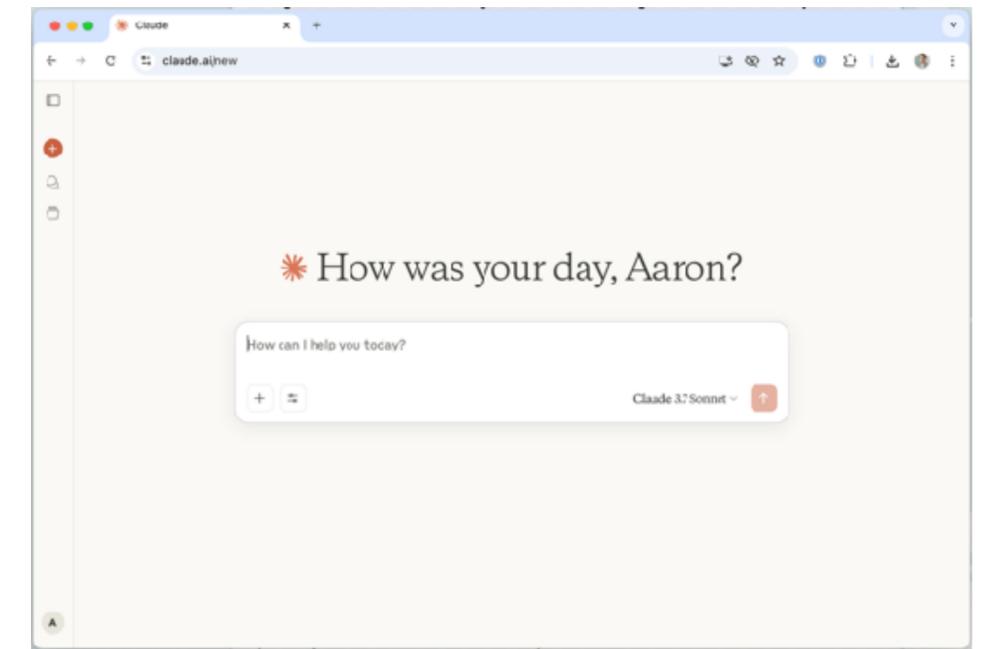
Logging in to an app with SSO



1. Login Screen



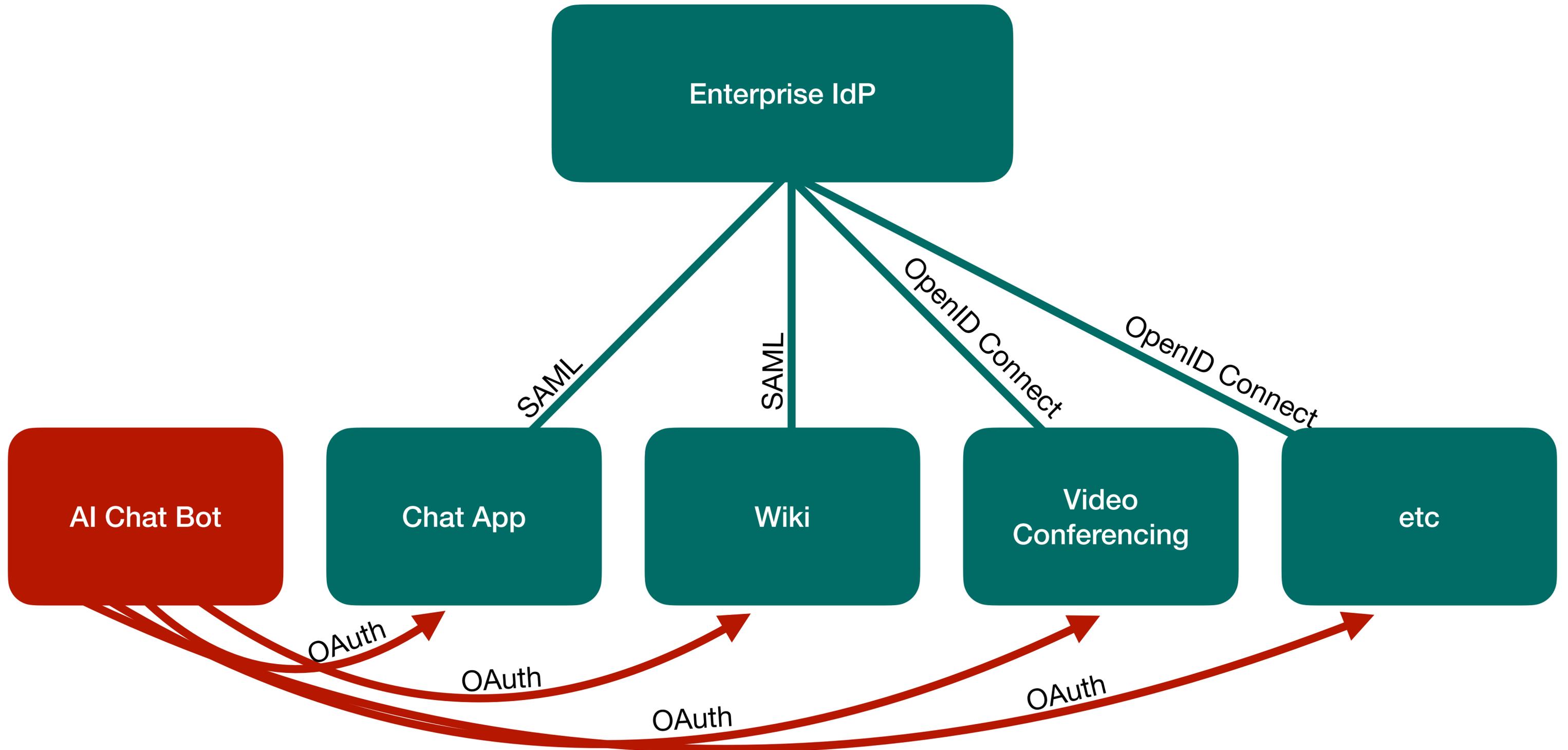
2. Redirect to IdP

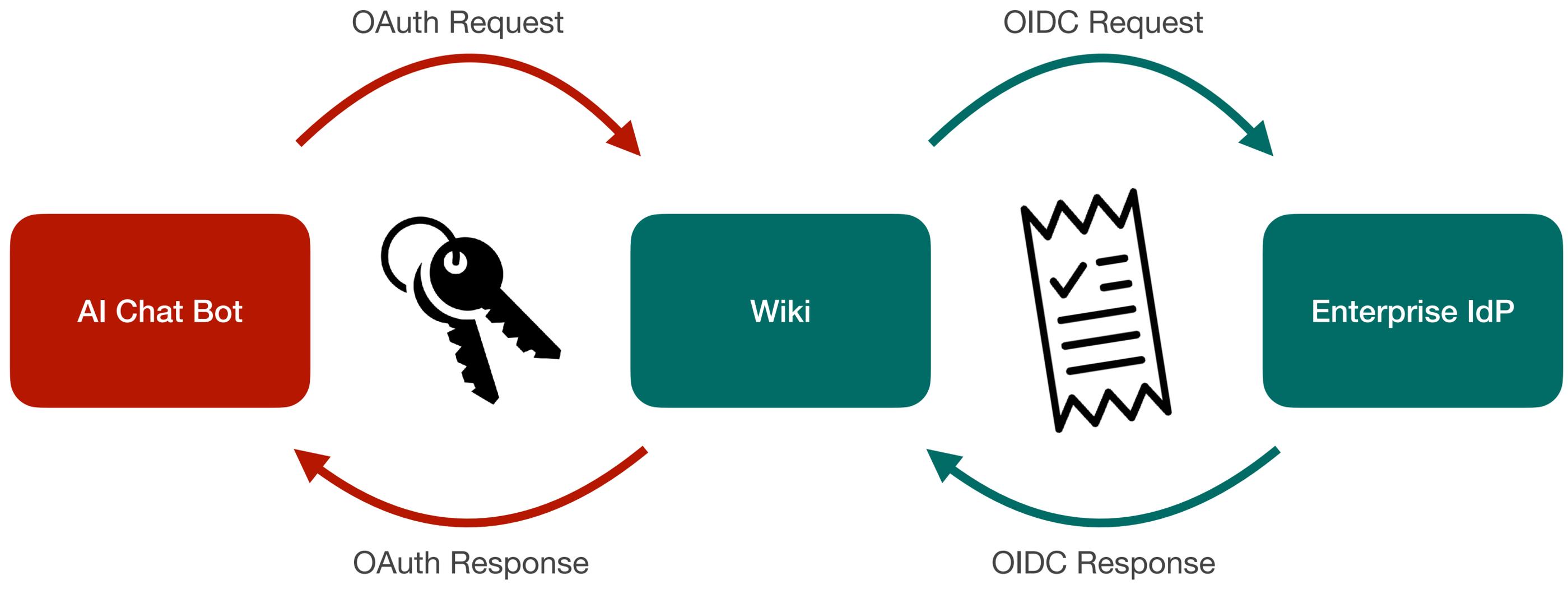


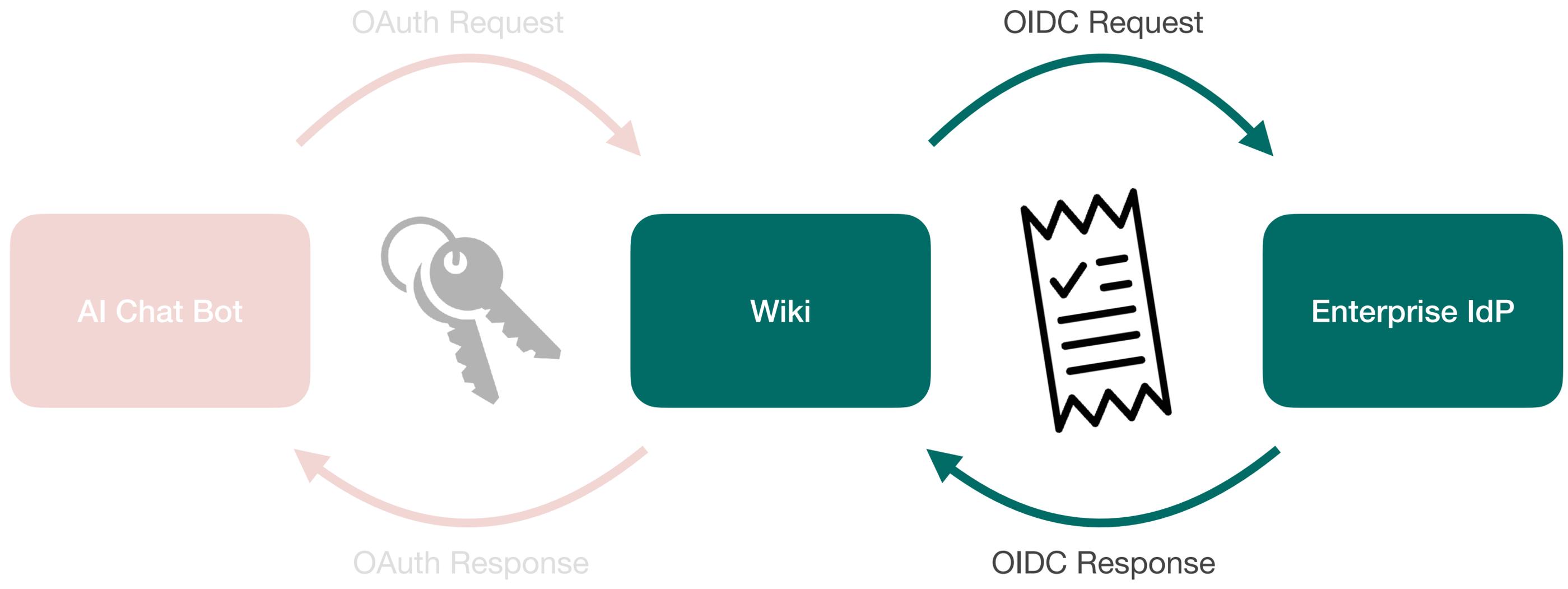
3. Logged In



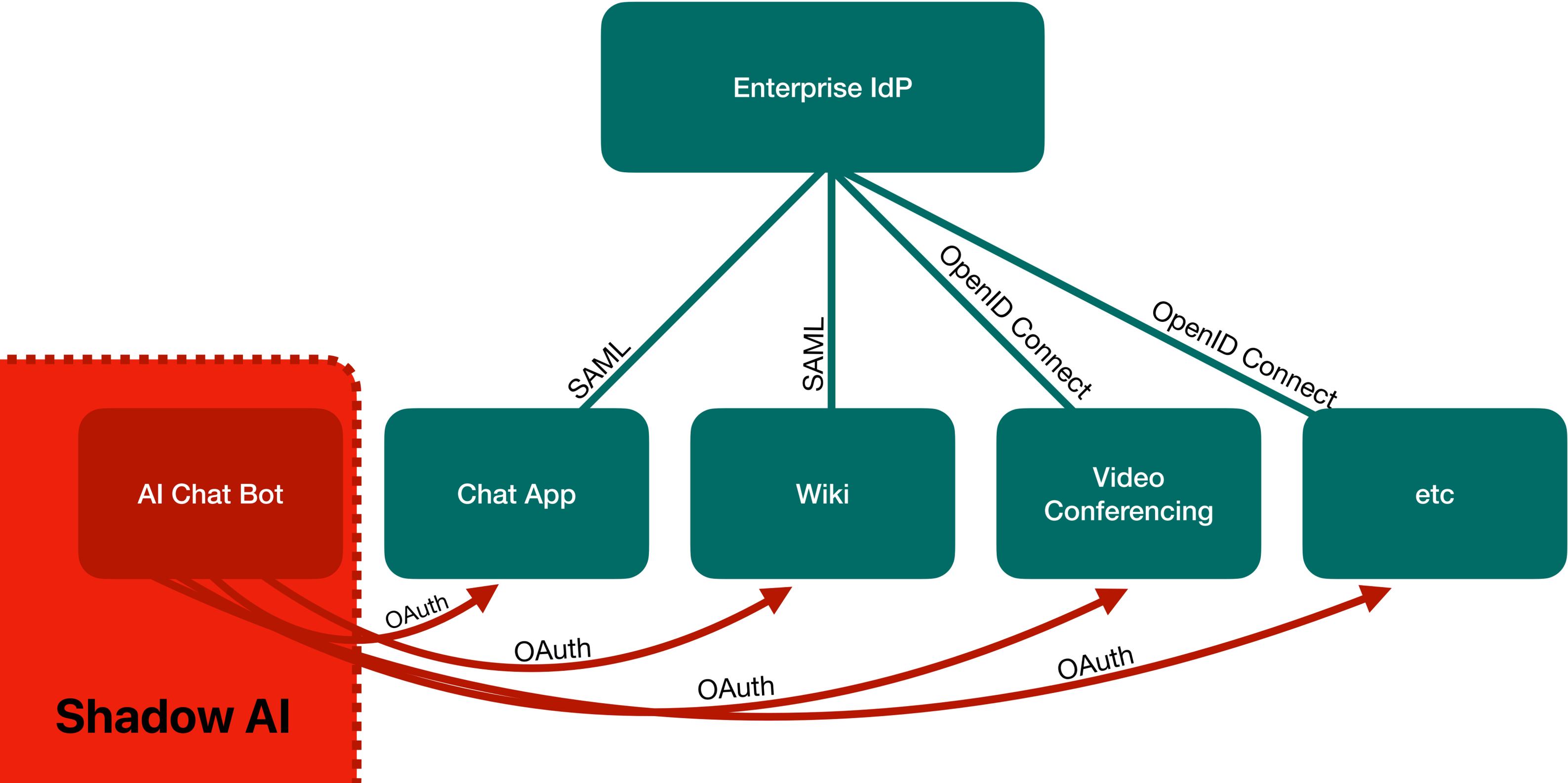
Enterprise API Access Today







Enterprise API Access Today



**No control or
visibility by the
enterprise
IT admin**

**Terrible UX
for Employees**

Cross App Access

aka Identity Assertion JWT Authorization Grant

Identity Assertion JWT Authorization Grant
draft-ietf-oauth-identity-assertion-authorization-grant-01

Status: IESG evaluation record | IESG writeups | Email expansions | History

Versions: 00 | 01

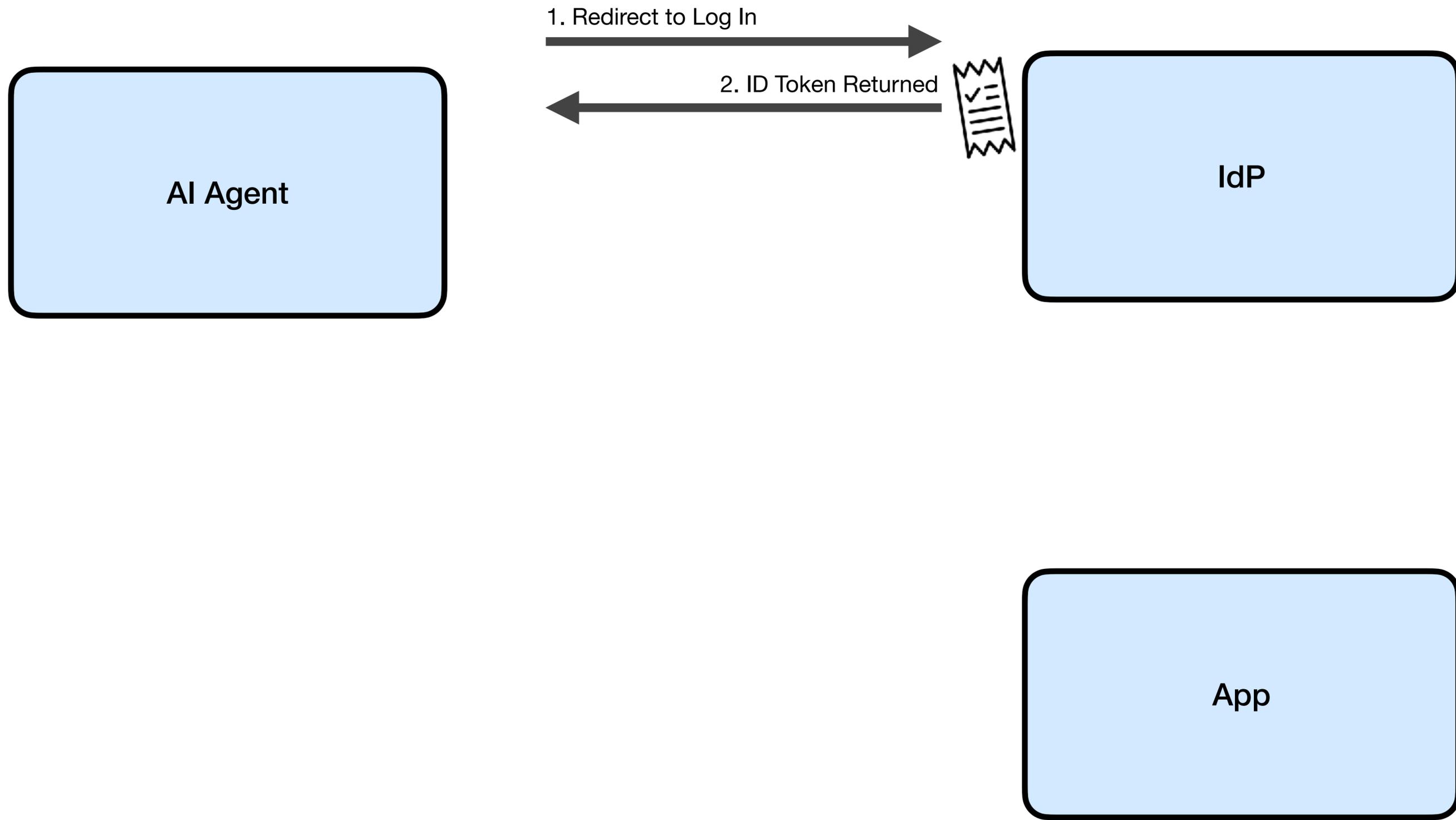
Timeline: Mar 2024 (00), Jul 2024 (01), Oct 2024 (02), Apr 2025 (03), Jun 2025 (0-05), Sep 2025 (00), Oct 2025 (01)

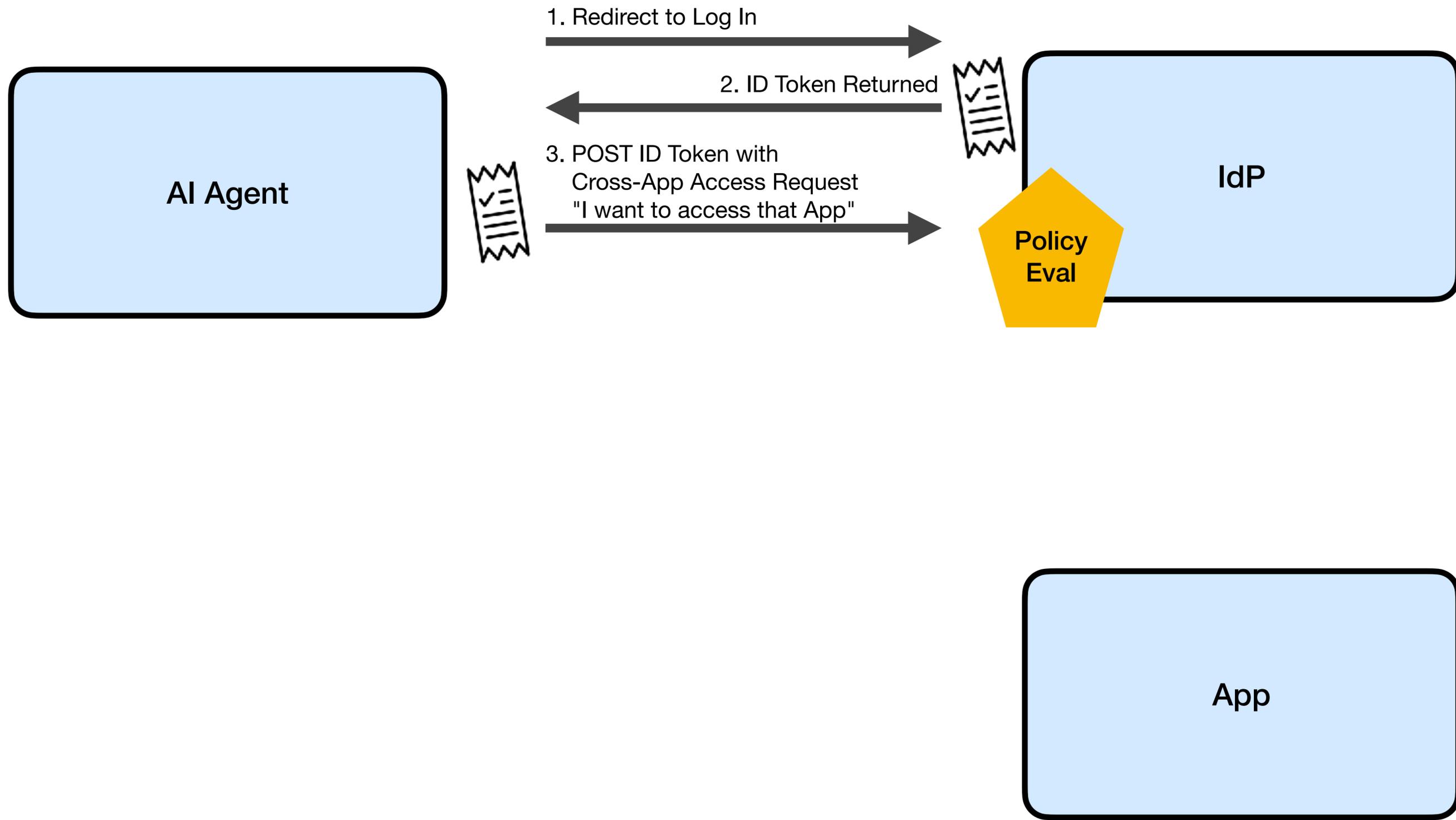
Document: Active Internet-Draft (oauth WG)
Authors: Aaron Parecki, Karl McGuinness, Brian Campbell
Last updated: 2025-10-19
Replaces: draft-parecki-oauth-identity-assertion-authorization-grant
RFC stream: Internet Engineering Task Force (IETF)
Intended RFC status: (None)
Formats: txt, html, xml, htmlized, pdf, bibtex, bibxml

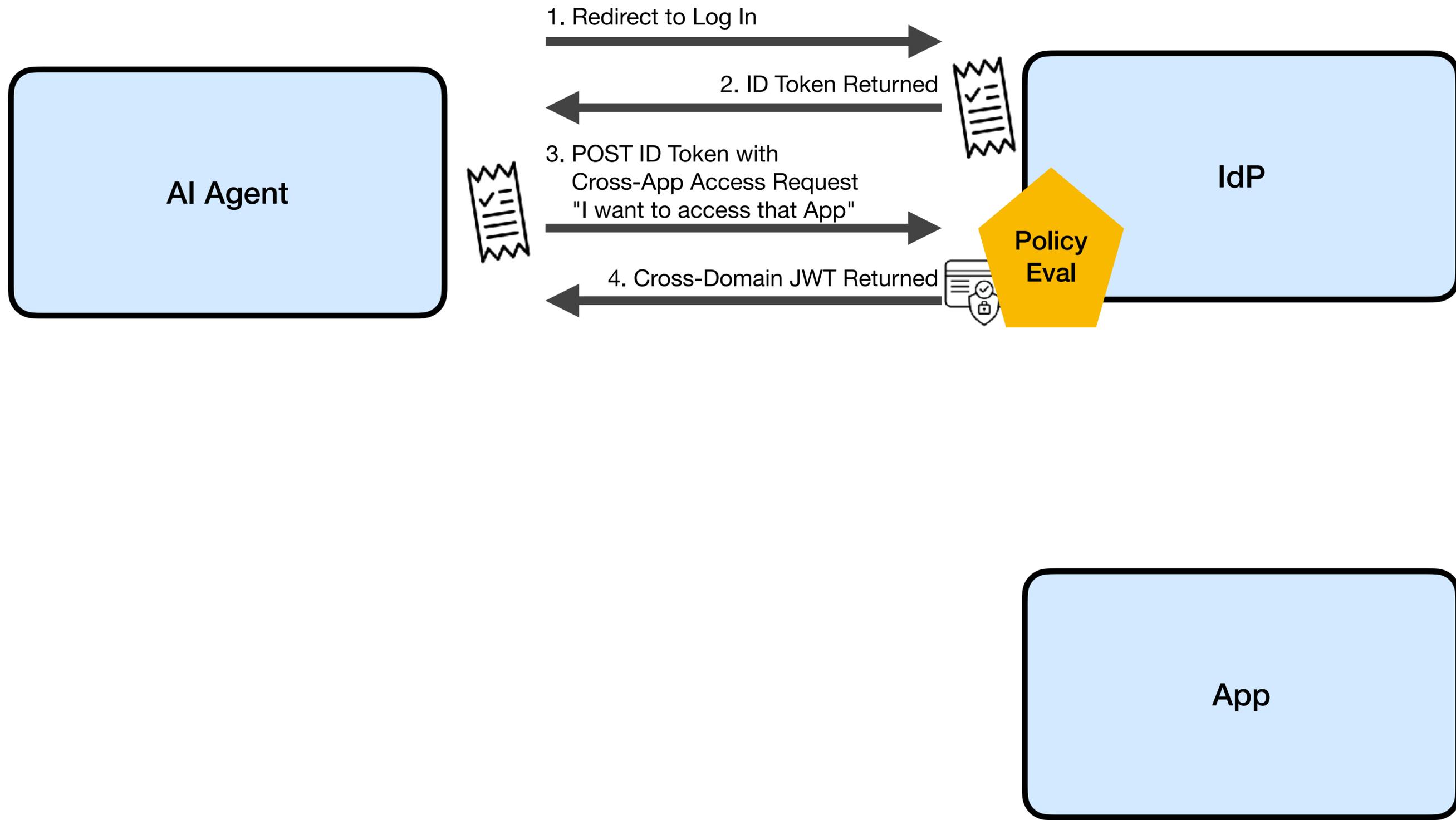
IETF®

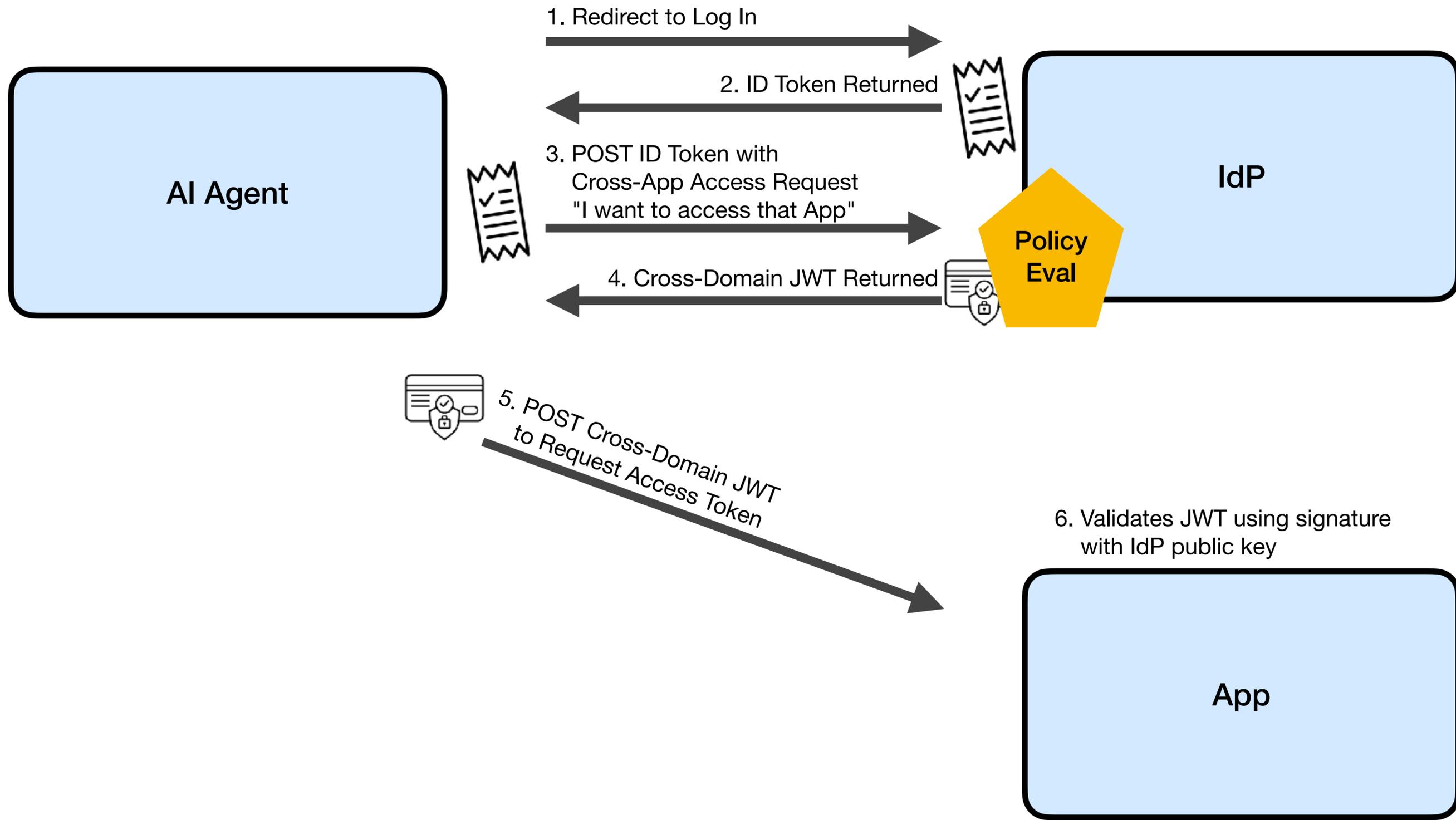
- Developed in the open
- First draft published March 2024
- Based on existing RFCs and work from many authors

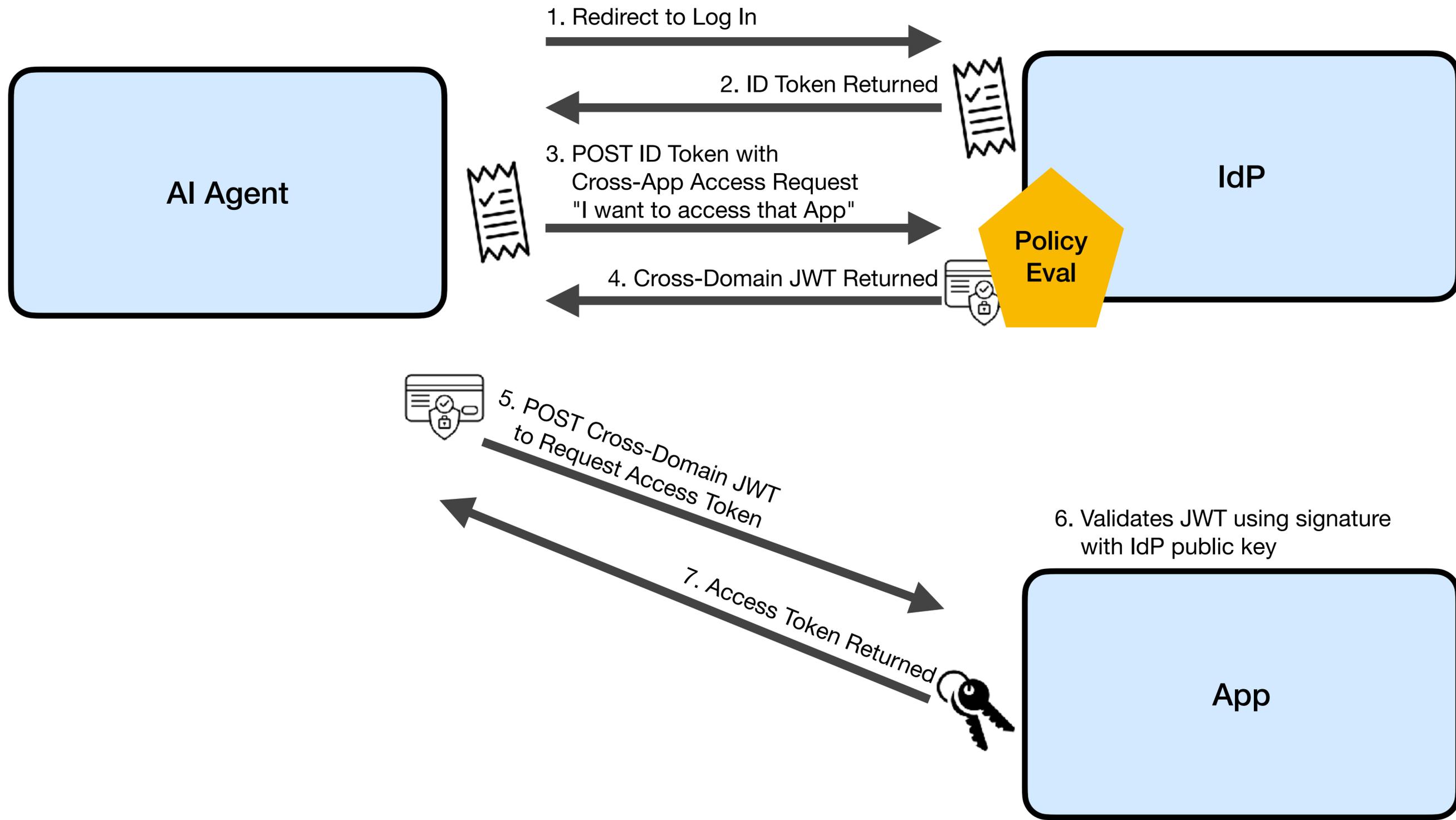




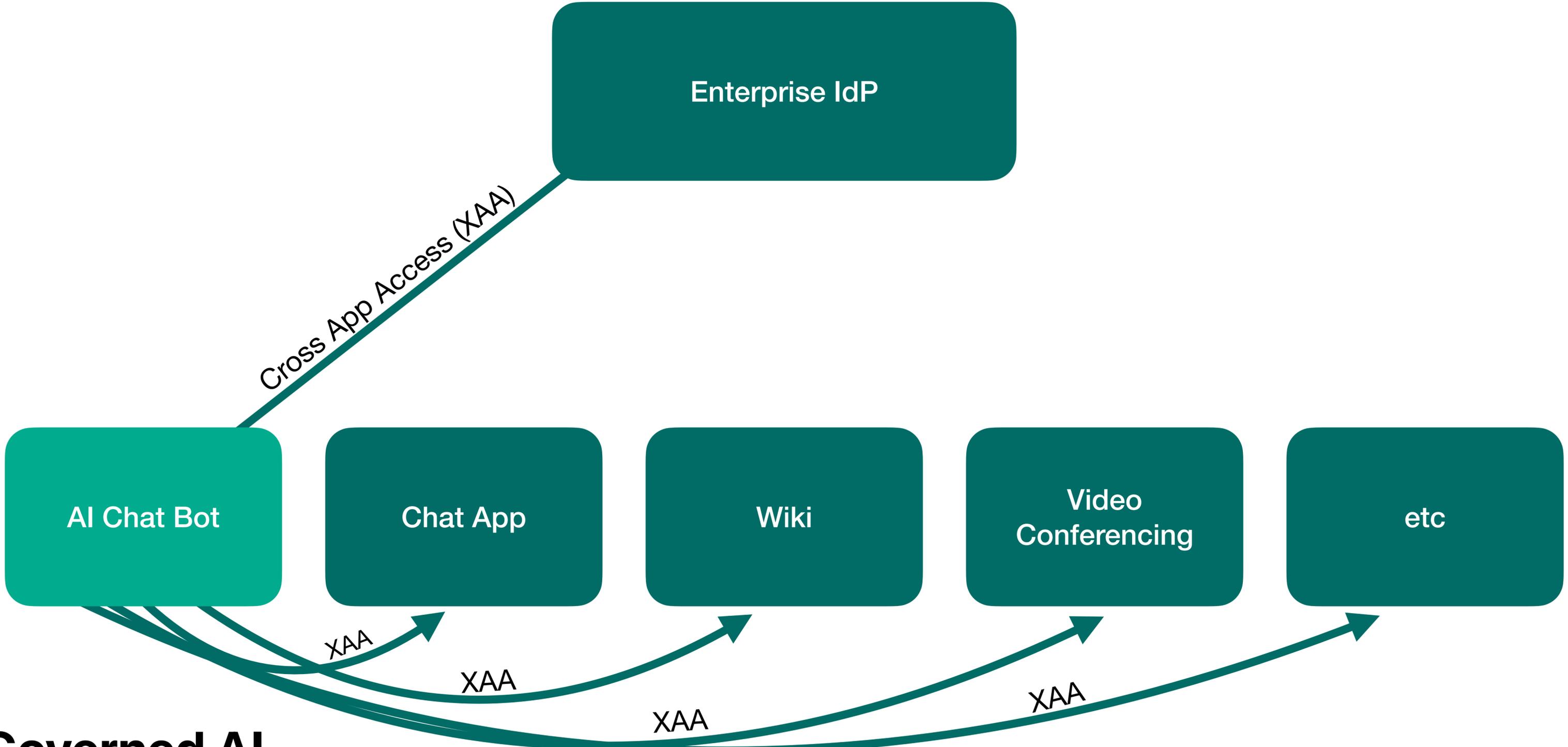




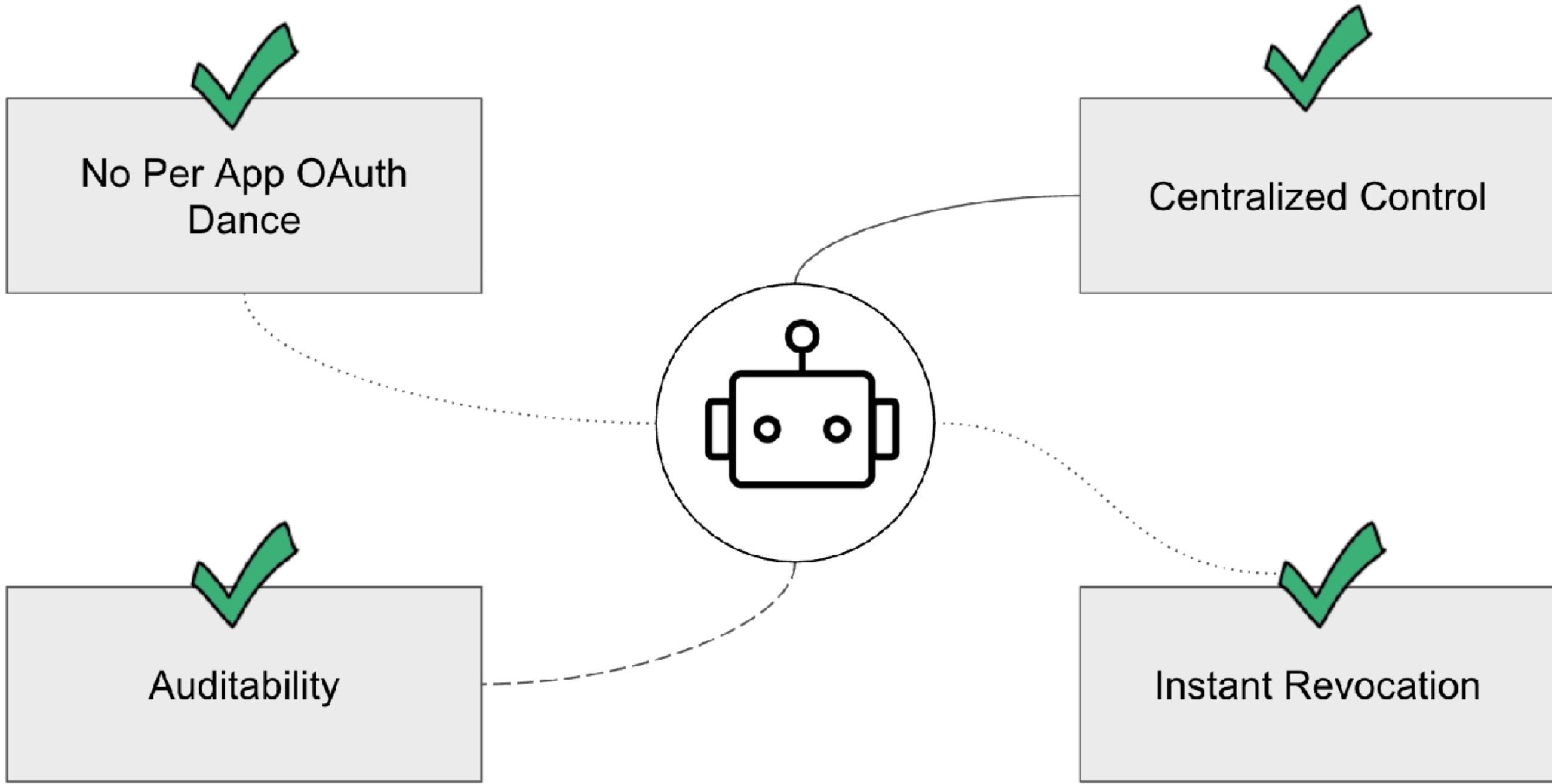




Cross App Access



Governed AI

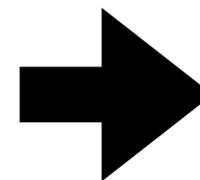




RFC 8693
OAuth Token Exchange



(Working Group Draft)
OAuth Identity and Authorization
Chaining Across Domains



Cross-App Access
(Working Group Draft)
Identity Assertion
JWT Authorization Grant

RFC 7523
JWT Profile for OAuth
Authorization Grants





Cross App Access (XAA)

The open sandbox for testing enterprise cross-application access flows

Test secure agent-to-app and app-to-app authorization on behalf of users using the [OAuth Identity Assertion Authorization Grant \(ID-JAG\)](#).

Try the XAA Flow →

What is Cross App Access (XAA) ?

Cross App Access (XAA) describes the use case where an application requires access to a third-party application, such as when your company's chat application displays the latest code update from a separate source code management tool. Using the OAuth extension "Identity Assertion Authorization Grant," the enterprise identity provider manages the connection between two applications, replacing the user's manual approval step with a seamless token exchange behind the scenes.

Explore the Roles



Requesting App

Configure and test applications that request access to protected resources using the XAA flow.



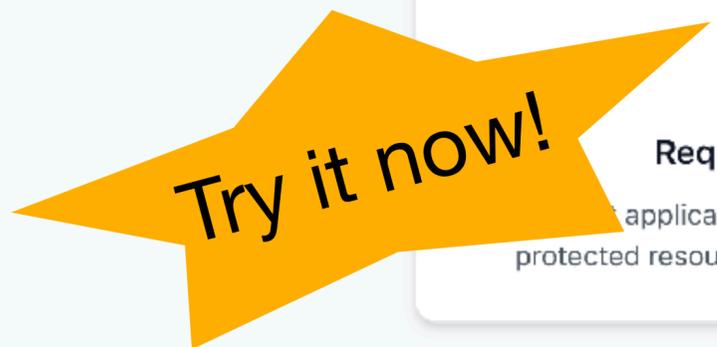
Resource App

Configure and test resource servers that validate XAA requests.



Identity Provider

Manage users and authenticate identities using the XAA flow.



<https://xaa.dev>