
Microsoft Authentication Library for JavaScript (MSAL.js)

The Microsoft Authentication Library for JavaScript enables both client-side and server-side JavaScript applications to authenticate users using Azure AD for work and school accounts (AAD), Microsoft personal accounts (MSA), and social identity providers like Facebook, Google, LinkedIn, Microsoft accounts, etc. through Azure AD B2C service. It also enables your app to get tokens to access Microsoft Cloud services such as Microsoft Graph.

Repository

Core, wrapper and extensions libraries

The `lib` folder contains the source code for our libraries in active development. You will also find all the details about **installing the libraries** in their respective README.md.

- Microsoft Authentication Library for Node.js: A Node.js library that enables authentication and token acquisition with the Microsoft Identity platform in JavaScript applications. Implements the following OAuth 2.0 protocols and is OpenID-compliant:
 - Authorization Code Grant with PKCE
 - Device Code Grant
 - Refresh Token Grant
 - Client Credential Grant
 - Silent Flow
 - On-behalf-of Flow
- Microsoft Authentication Library for JavaScript: A browser-based, framework-agnostic browser library that enables authentication and token acquisition with the Microsoft Identity platform in JavaScript applications. Implements the OAuth 2.0 Authorization Code Flow with PKCE, and is OpenID-compliant.
- Microsoft Authentication Library for React: A wrapper of the msal-browser library for apps using React.
- Microsoft Authentication Library for Angular: A wrapper of the msal-browser library for apps using Angular framework.
- Microsoft Authentication Extensions for Node: The Microsoft Authentication Extensions for Node offers secure mechanisms for client applications to perform cross-platform token cache serialization and persistence. It gives additional support to the Microsoft Authentication Library for Node (MSAL).

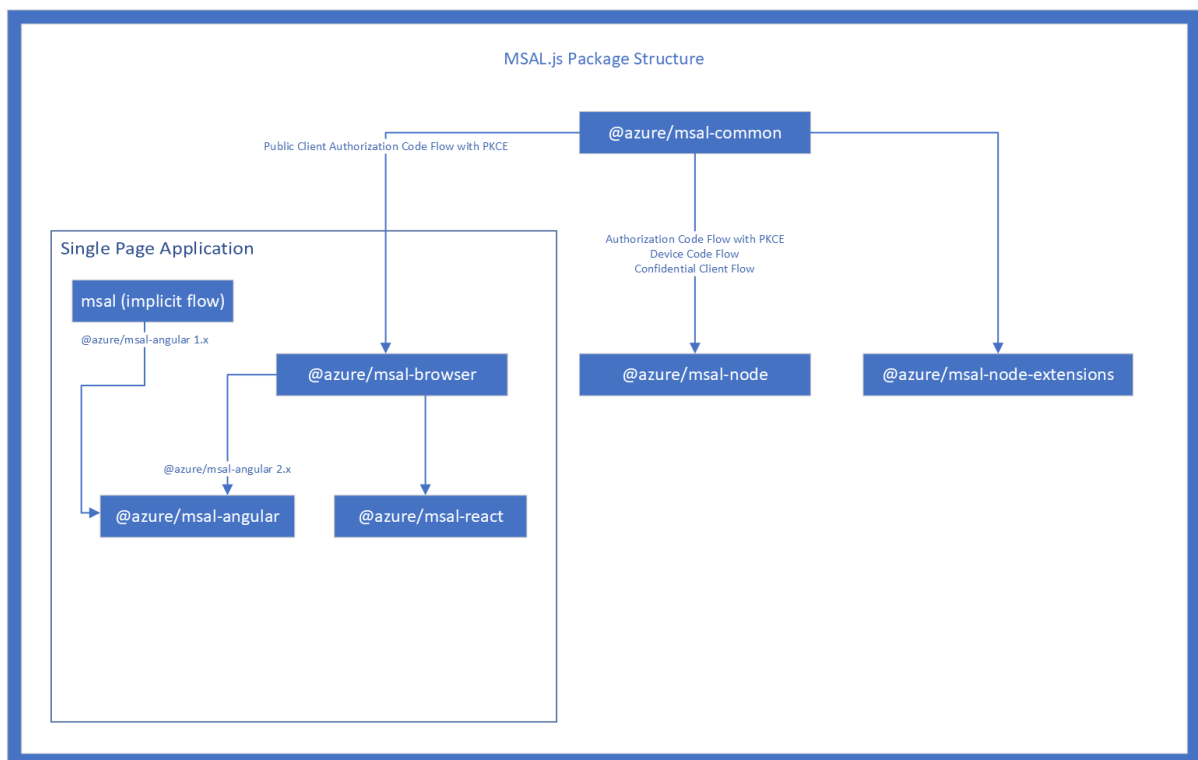
Libraries in Long-term Support (LTS)

The following libraries, hosted on the [msal-lts](#) branch, are no longer in active development, but they are still receiving critical security bug fix support.

- Microsoft Authentication Library for JavaScript v2.x
- Microsoft Authentication Library for Node.js v1.x
- Microsoft Authentication Library for React v1.x
- Microsoft Authentication Library for Angular v2.x

Package Structure

We ship a number of different packages which are meant for different platforms. You can see the relationship between packages and different authentication flows they implement below.



Samples

The [samples](#) folder contains sample applications for our libraries. A complete list of samples can be found in the respective package folders or on our wiki.

Package versioning

All of our libraries follow semantic versioning. We recommend using the latest version of each library to ensure you have the latest security patches and bug fixes.

Roadmap

Please check the roadmap to see what we are working on and what we have planned for future releases.

Community Help and Support

- GitHub Issues is the best place to ask questions, report bugs, and new request features.
- FAQs for access to our frequently asked questions.
- Stack Overflow using “msal” and “msal.js” tag.

Contribute

We enthusiastically welcome contributions and feedback. Please read the contributing guide before you begin.

Security Reporting

If you find a security issue with our libraries or services please report it to the Microsoft Security Response Center (MSRC) with as much detail as possible. Your submission may be eligible for a bounty through the Microsoft Bounty program. Please do not post security issues to GitHub Issues or any other public site. We will contact you shortly upon receiving the information. We encourage you to get notifications of when security incidents occur by visiting this page and subscribing to Security Advisory Alerts.

License

Copyright (c) Microsoft Corporation. All rights reserved. Licensed under the MIT License (the “License”).

We Value and Adhere to the Microsoft Open Source Code of Conduct

This project has adopted the Microsoft Open Source Code of Conduct. For more information see the Code of Conduct FAQ or contact opencode@microsoft.com with any additional questions or comments.