## Vendor:

Blue Button 2.0

**Website Link:** https://bluebutton.cms.gov/developers/

## API Description:

The Centers for Medicare and Medicaid Services (CMS) Blue Button API enables Medicare beneficiaries to connect their Medicare claims data to the applications, services, and research programs they trust.

The CMS Blue Button API:

- Enables a developer to register a beneficiary-facing application
- Enables a beneficiary to grant an application access to four years of their Part A, B, and D claims data
- Uses the HL7 FHIR standard for beneficiary data and the OAuth 2.0 standard for beneficiary authorization

## Important Information:

**Authorization**

To use the Blue Button OAuth 2 a developer must register their application.

A registered application is given a client ID and a client secret. The secret should only be used if it can be kept confidential, such as communication between your server and the Blue Button API. Otherwise the Client Application Flow may be used.

**Native Mobile App Support**

Native Mobile App Support follows the RFC 8252 - OAuth 2.0 for Native Apps authentication flow utilizing the PKCE extension and enables a custom URI scheme redirect.

The implementation of the RFC 8252 specification enables developers to build mobile applications without requiring a proxy server to route redirect calls to their mobile app.

The PKCE extension provides a technique for public clients to mitigate the threat of a "man-in-the-middle" attack. This involves creating a secret that is used when exchanging the authorization code to obtain an access token.

PKCE uses a code challenge that is derived from a code-verifier. The standard supports two styles of code challenge:

- plain
- S256

However, Blue Button 2.0 only supports the "S256" style code challenge.

Where the:

codechallenge = BASE64URL-ENCODE(SHA256(ASCII(codeverifier)))

The following additional parameters and values are sent as part of the OAuth2.0 Authorization Request:

- code_challenge
- codechallengemethod = "S256"

More details can be found about this flow on OAuth.com. Check out this link: Protecting Mobile Apps with PKCE - OAuth 2.0 Servers




Connect with attendees on Twitter by sharing your experience of the Connectathon happenings. Use #MiConnectathon in your tweet to make it easier to connect with attendees, MiHIN and the world!