

Microsoft.IdentityModel.Protocol.Extensions

August 18, 2026

Microsoft.IdentityModel.Protocol.Extensions

Kind: Service

Source: Pams/API/Pams.API/Pams.API.csproj (line 1)

Part of: Pams

NuGet package dependency

Pams.API references Microsoft.IdentityModel.Protocol.Extensions as a NuGet dependency for identity protocol extension types used during token and protocol processing. The package runs within the API project and is not consumed directly by JavaScript clients.

Diagram

```
sequenceDiagram
    participant Client
    participant PamsAPI as Pams.API
    participant IdentityExtensions as Microsoft.IdentityModel.Protocol.Extensions

    Client->>PamsAPI: Send request with bearer token
    PamsAPI->>IdentityExtensions: Process identity protocol data
    IdentityExtensions-->>PamsAPI: Return protocol extension results
    PamsAPI-->>Client: Return API response
```

Usage

```
async function getPamsData(accessToken: string) {
    const response = await fetch("/api/pams", {
        headers: {
            Authorization: `Bearer ${accessToken}`,
            Accept: "application/json",
        },
    });
};
```

```

    if (!response.ok) {
        throw new Error(`Pams.API request failed: ${response.status}`);
    }

    return response.json();
}

// Microsoft.IdentityModel.Protocol.Extensions is used by Pams.API
// while processing the authenticated request.
const data = await getPamsData("access-token");
console.log(data);

```

AI Coding Instructions

- Keep `Microsoft.IdentityModel.Protocol.Extensions` referenced through the `Pams.API.csproj` NuGet dependency rather than adding JavaScript client-side package usage.
- Treat identity protocol processing as server-side behavior; clients should send authenticated requests to `Pams.API`.
- Check package version compatibility with other `Microsoft.IdentityModel` dependencies before changing the project reference.
- Avoid handling or logging bearer tokens in client code beyond sending them in the `Authorization` header.

Used by

1 reference from 1 file. Each is a place in this repository where the symbol is actually used — go read one rather than trusting an example.

Injected or called by (1)

- `Pams.API` — `Pams/API/Pams.API/Pams.API.csproj :1`