# Microsoft.Owin.Security.OAuth

August 17, 2026

## Microsoft.Owin.Security.OAuth

**Kind:** Service

**Source:** `Pams/Core/Pams.Security/Pams.Security.csproj` (line 1)

**Part of:** Pams

NuGet package dependency

`Microsoft.Owin.Security.OAuth` is a NuGet dependency in `Pams.Security` that provides OAuth authentication middleware for OWIN-based .NET applications. It supports issuing, validating, and processing bearer tokens for protected application endpoints.

### Diagram

```
sequenceDiagram
    participant Client
    participant TokenEndpoint
    participant OAuthMiddleware
    participant ProtectedApi

    Client->>TokenEndpoint: Submit OAuth credentials
    TokenEndpoint->>OAuthMiddleware: Validate grant request
    OAuthMiddleware-->>Client: Return access token
    Client->>ProtectedApi: Send request with bearer token
    ProtectedApi->>OAuthMiddleware: Validate bearer token
    OAuthMiddleware-->>ProtectedApi: Return authenticated identity
    ProtectedApi-->>Client: Return protected response
```

### Usage

```
async function getProtectedData(username: string, password: string) {
  const tokenResponse = await fetch("/token", {
    method: "POST",
    headers: {
      "Content-Type": "application/x-www-form-urlencoded",
```

```
    },
    body: new URLSearchParams({
      grant_type: "password",
      username,
      password,
    }),
  });

  if (!tokenResponse.ok) {
    throw new Error("Token request failed");
  }

  const token = await tokenResponse.json();

  const response = await fetch("/api/protected-resource", {
    headers: {
      Authorization: `Bearer ${token.access_token}`,
    },
  });

  if (!response.ok) {
    throw new Error("Protected API request failed");
  }

  return response.json();
}
```

## AI Coding Instructions

- Keep OAuth server and bearer authentication configuration in the OWIN startup pipeline.
- Match the token endpoint path and grant types expected by the `Pams.Security` host application.
- Send access tokens through the `Authorization: Bearer` request header when calling protected endpoints.
- Avoid exposing client secrets, passwords, or access tokens in browser storage, logs, or error messages.