

야후 OAuth 2.0 로그인

Middleware that enables an application to support Yahoo Japan's OAuth 2.0 authentication workflow.

How To Use

```
// Startup.cs
app.UseYahooJapanAuthentication(
    new YahooJapanAuthenticationOptions
    {
        ClientId = "ClientId",
        ClientSecret = "ClientSecret",
        SignInAsAuthenticationType = DefaultAuthenticationTypes.ExternalCookie
    }
);
```

Owin Authentication Flow

AuthenticationMiddleware

CreateHanlder()에서 AuthenticationHandler 인스턴스 생성

AuthenticationOptions의 기본 값들 설정

FLOW

1. SNS 로그인 버튼 클릭

인증 관리자에 대한 Challenge 생성 → 응답의 상태 코드를 401로 변경

Challenge : 신원을 확인해 달라는 요청

```
context.HttpContext.GetOwinContext().Authentication.Challenge(properties, LoginProvider);
```

2. AuthenticationHandler.ApplyResponseChallengeAsync()

주로 401 상태 코드의 응답(Challenge)을 처리하기 위한 로직 구현

Resource Owner와 상호 작용하고 보호된 리소스에 액세스할 수 있는 권한을 얻기 위해 Authorization Endpoint로 redirect

Resource Owner는 로그인 후 리소스 사용에 동의

3. AuthenticationHandler.InvokeAsync()

Authorization Server의 올바른 인증 콜백인지 확인한다.

올바른 인증 콜백이면 AuthenticateAsync 메소드를 호출해서 인증 결과를 반환한다.

```
AuthenticationTicket ticket = await AuthenticateAsync();
```

4. AuthenticationHandler.AuthenticateCoreAsync()

state와 code 파라미터 값으로 validation check

code 값을 이용해 TokenEndPoint에 요청해서 access token을 발급받는다.

access token을 사용해 리소스 요청(계정 정보)

```
// REQUEST
// nonce 생성
GenerateCorrelationId(properties);
// state 생성
var state = Options.StateDataFormat.Protect(properties);

// RESPONSE
// state 값 쿼리스트링에서 가져오기
var state = GetValue(query, "state");
properties = Options.StateDataFormat.Unprotect(state);
if (properties == null)
{
    return null;
}
// 검증
if (!ValidateCorrelationId(properties, _logger))
{
    return new AuthenticationTicket(null, properties);
}
```

5. 얻은 정보를 DB에 넣어 회원가입 또는 로그인

Code

주요 코드 링크

Result



<https://www.adprint.jp/Members/LoginView>

Reference

<https://jjoystory.tistory.com/45>

<https://jjoystory.tistory.com/46>

<https://jjoystory.tistory.com/47>