Bearer

Bearer 토큰을 전달하는 방법

OAuth Bearer 토큰의 사용법을 정의한 스펙에서는 세 가지 방법의 토큰 전달 방법을 설명하고 있다.

- HTTP Authorization 헤더를 이용하는 방법
- 폼 인코딩된 요청 파라미터로 전달하는 방법
- URL 인코딩된 질의 파라미터로 전달하는 방법

다른 두 가지 방법은 한계가 있기 때문에 가능하면 AUthorization 헤더를 이용하는 방법을 권장하고 있다. 질의 파라미터로 액세서 토큰을 전달할 때 엑세스 토큰이 URL 질의에 포함되기 때문에 실수로 서버의 로그에 액세스 토큰 값이 유출될 수 있다. 폼 인코딩된 파라미터로 액세스 토큰을 전달하는 방법은 보호된 리소스의 입력 유형이 폼 인코딩된 파라미터로 제한되고 POST 전송만 사용할 수 있다. 만약, API가 이미 그렇게 사용하도록 정해졌다면, 질의 파라미터로 전달할 때 발생할 수 있는 동일한 보안 이슈가 발생하지 않으므로 문제되지 않는다.

Authorization 헤더를 이용하는 방법은 세 가지 방법 중에서 갖아 큰 유연성과 보안성을 제공하지만, 일부 클라이언트에서는 사용하기 어려울 수 있다는 단점이 있다. 견고하게 작성된 클라이언트나 서버 라이브러리는 세 가지 방법 모두 제공할 것이다.

Bearer 토큰

기술적인 관점에서 보면, Bearer 토큰은 웹 브라우저 쿠키와 매우 유사하다고 생각할 수 있다. 기본적으로 웹 브라우저 쿠키와 동일한 점은 다음과 가다.

- 평문으로 된 문자열을 사용
- 시크릿이나 시그니처를 포함하지 않음
- 보안 모델에 있어서 TLS는 기본

차이점은 다음과 같다.

- 웹 브라우저에서는 쿠키를 오래 전부터 사용해왔지만 OAuth 클라이언트에서는 그렇지 않다.
- 웹 브라우저에서는 동일 근원 정책을 강제하기 때문에 어느 한 도메인을 위한 쿠키가 다른 도메인으로 전달되지 않는다. 하지만 OAuth 클라이언트의 경우에는 그렇지 않다

Bearer 토큰 사용에 있어서의 위험과 고려 사항

OAuth의 Bearer 토큰과 관련된 보안 위협은 다음과 같다.

- 토큰 위조: 공격자가 자신만의 가짜 토큰을 만들거나 기존의 유효한 토큰을 변조해 리소스 서버가 클라이언트에게 접근 권한을 인가하게 만들 수 있다. 예를 들면, 기존에는 접근할 수 없었던 정보에 접근하기 위해 공격자는 토큰을 인위적으로 만들 수 있다. 또는 토큰을 변조하거나 토큰 자체의 유효성을 확장할 수 있다.
- 토큰 재사용: 공격자는 이미 과거에 사용돼 유효 기간이 만료된 토큰을 이용하려고 시도한다. 그런 경우, 리소스 서버는 어떤 유효한 데이터도 반환을 해주면 안 되고, 대신 에러를 반환해야 한다. 구체적인 시나리오를 들어보면, 공격자는 처음에는 합법적으로 액세스 토큰을 획득한 후, 해당 토큰이 만료된 이후에 그것을 재사용하려고 시도하는 것이다.
- 토큰 리다이렉트: 공격자는 특정 리소스 서버를 위해 만들어진 토큰을 이용해 다른 리소스 서버에 접근하려고 시도한다. 그 경우, 공격 대상 리소스 서버는 공격자가 사용한 토큰이 유효한 것이라고

잘못 판단하는 경우다. 즉, 공격자는 특정 리소스 서버를 위한 액세스 토큰을 합법적으로 취득한 후, 그 토큰을 이용해 다른 리소스 서버에 접근하려고 시도하는 것이다.
- 토큰 유출: 토큰에는 시스템에 대한 민감한 정보가 포함될 수 있고, 공격자는 다른 방법으로는 획득할 수 없는 정보를 토큰을 통해 획득할 수 있다.

관련 문서

- OAuth2 in Action

From:
https://jace.link/ - **Various Ways**

Permanent link:
https://jace.link/open/bearer

Last update: **2020/06/02 09:25**