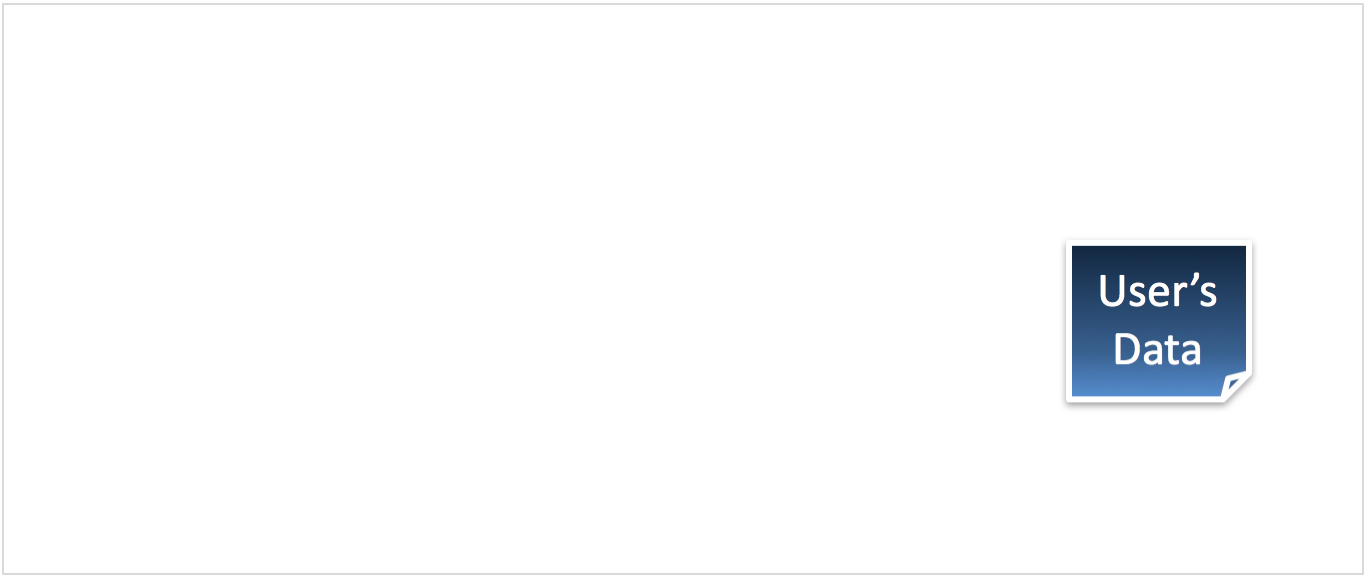
The Simplest Guide To OAuth 2.0

Takahiko Kawasaki Follow

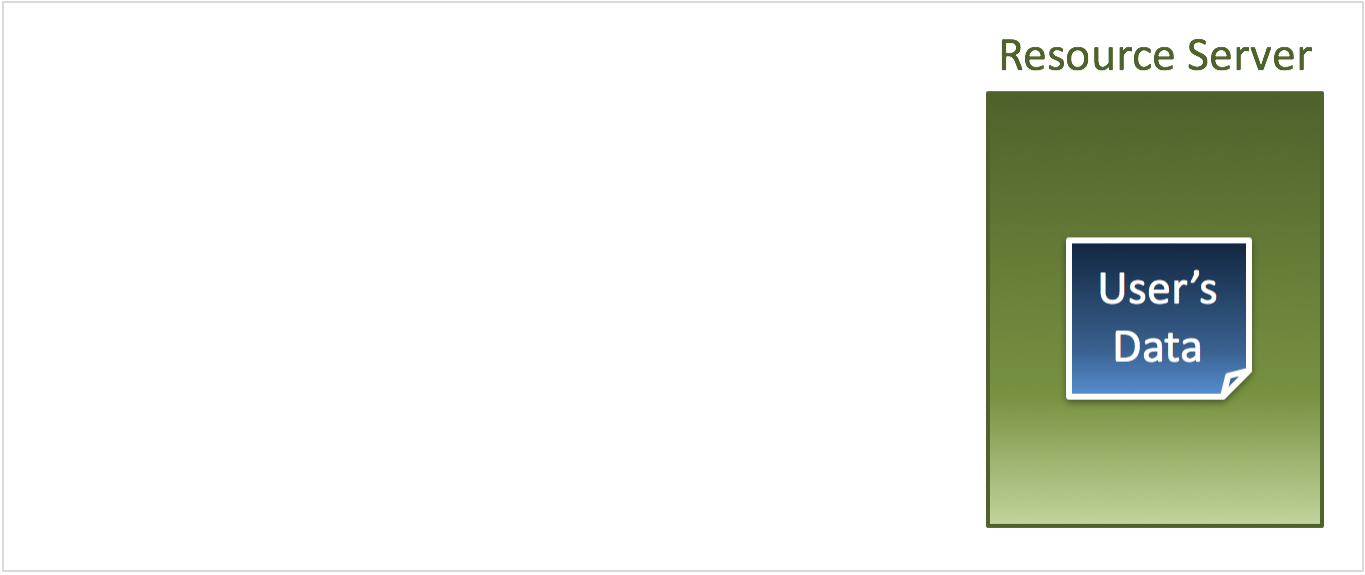
Aug 1, 2017 · 6 min read

For the past three years, I've repeated to explain OAuth 2.0 to those who don't have a technical background, mainly to investors as a co-founder of Authlete, Inc. (Tech In Asia: *"API security startup Authlete raises $1.2m in seed funding"*). As a result, I found a way to explain OAuth 2.0 in an easily understandable manner. This article introduces the steps.

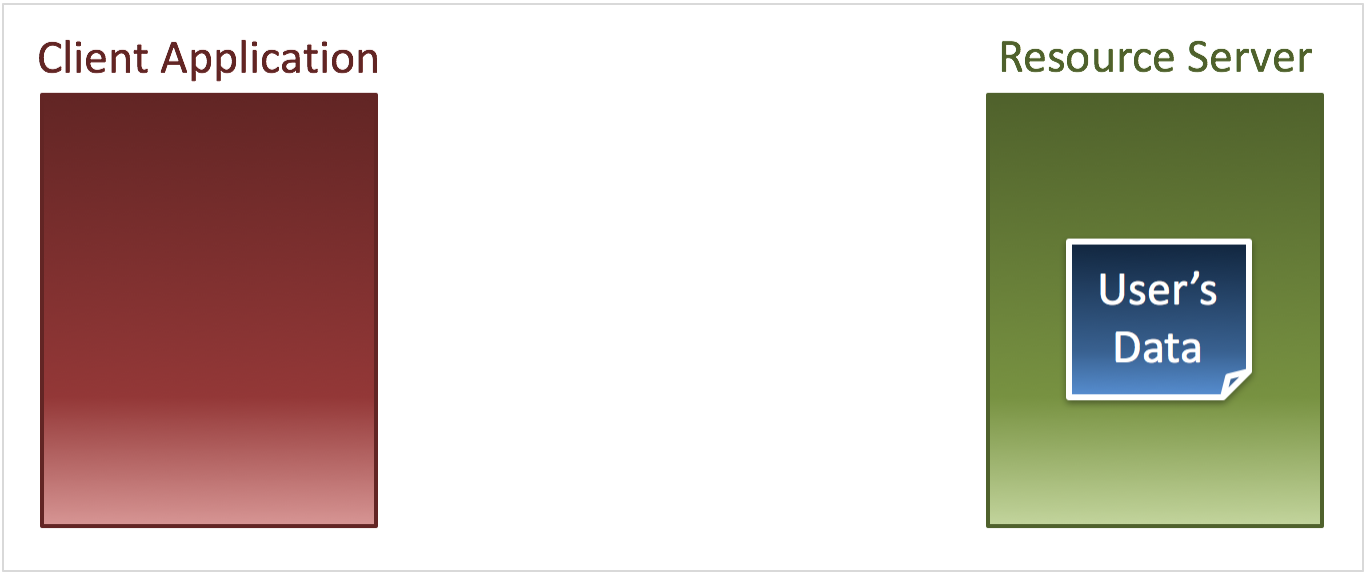
1. There are data of a user.



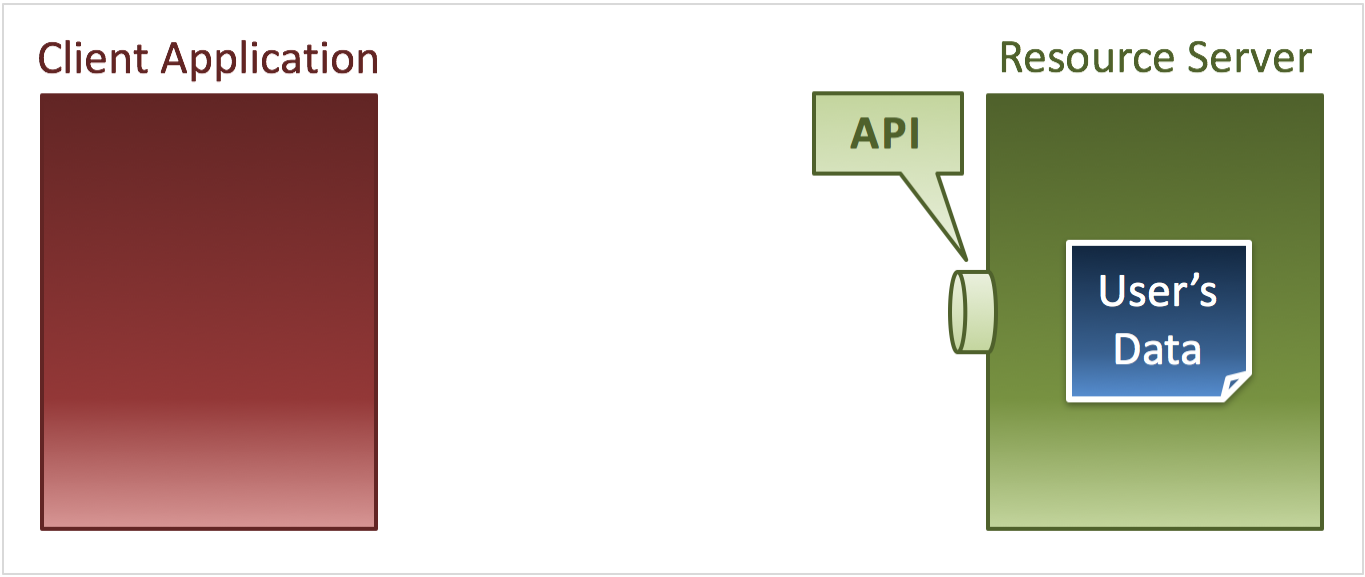
2. There is a server which manages the user's data. The server is called "Resource Server".



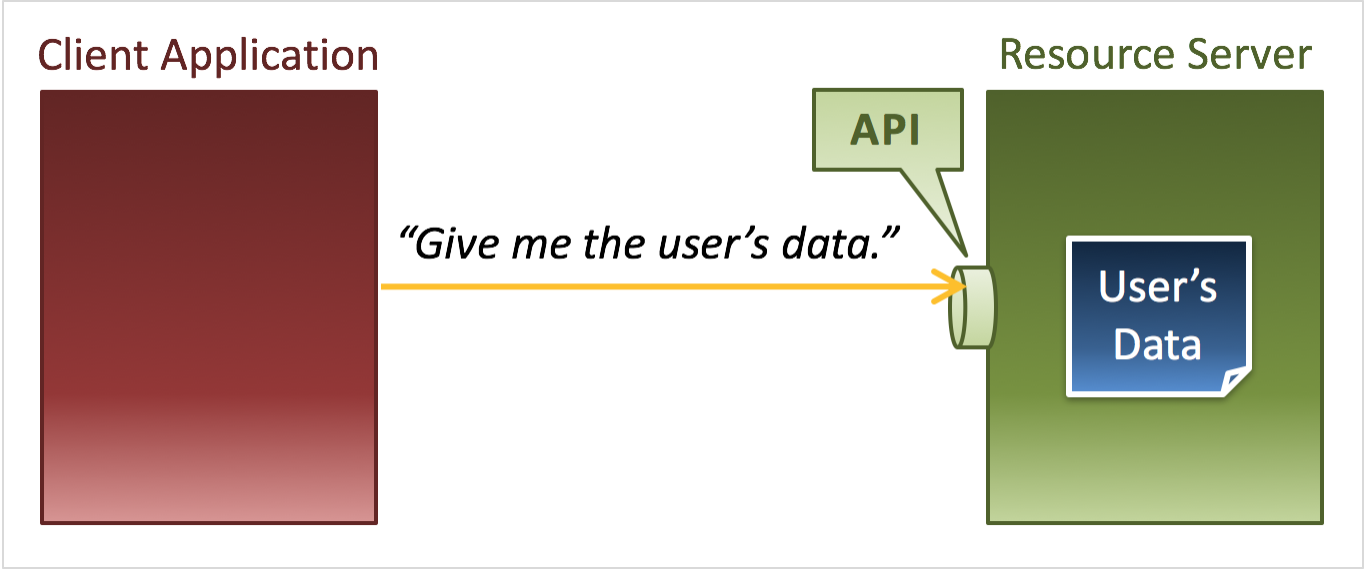
3. There is a "Client Application" which wants to use the user's data.



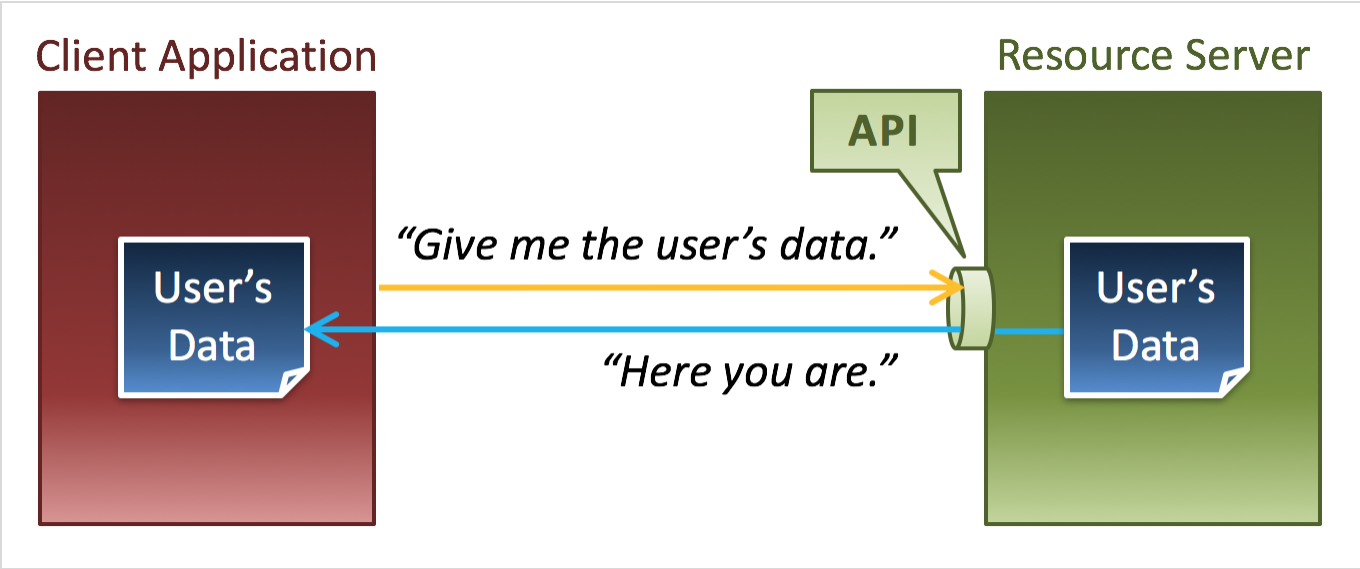
4. Let's prepare a gate to pass the user's data through. The gate is called "API".



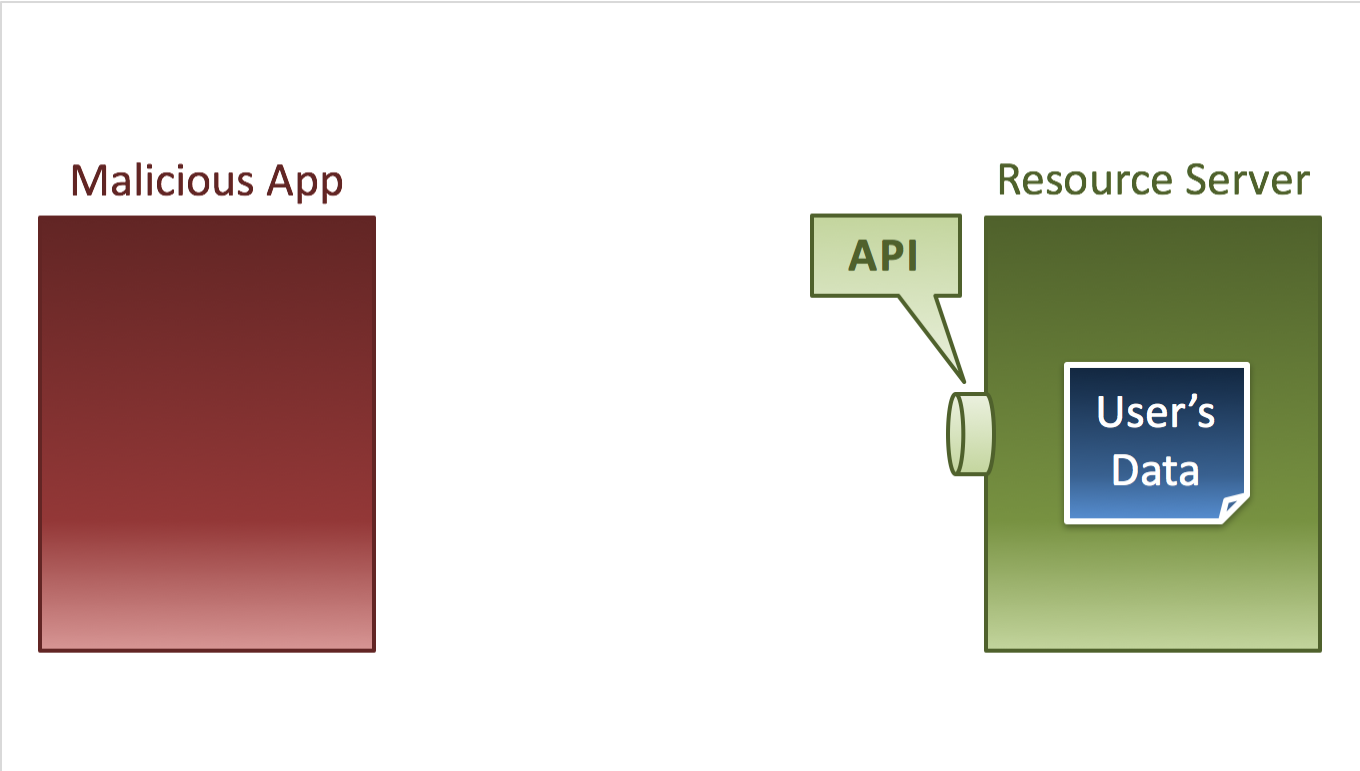
5. The client application requests the user's data.



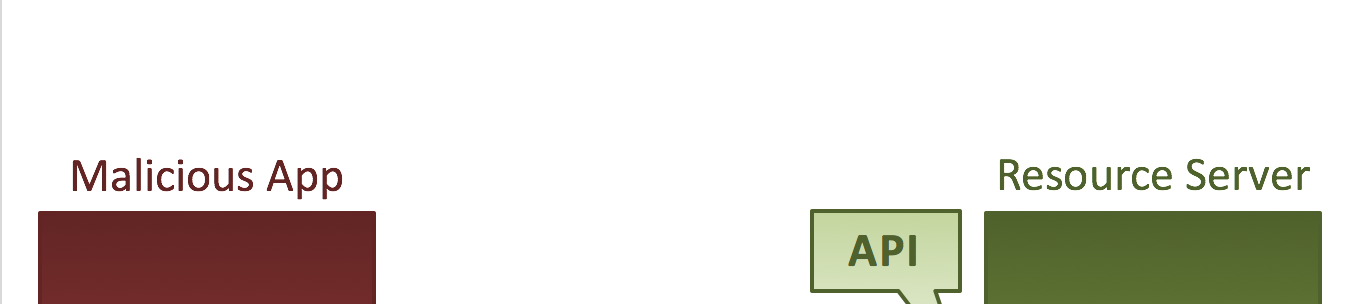
6. The resource server returns the user's data.

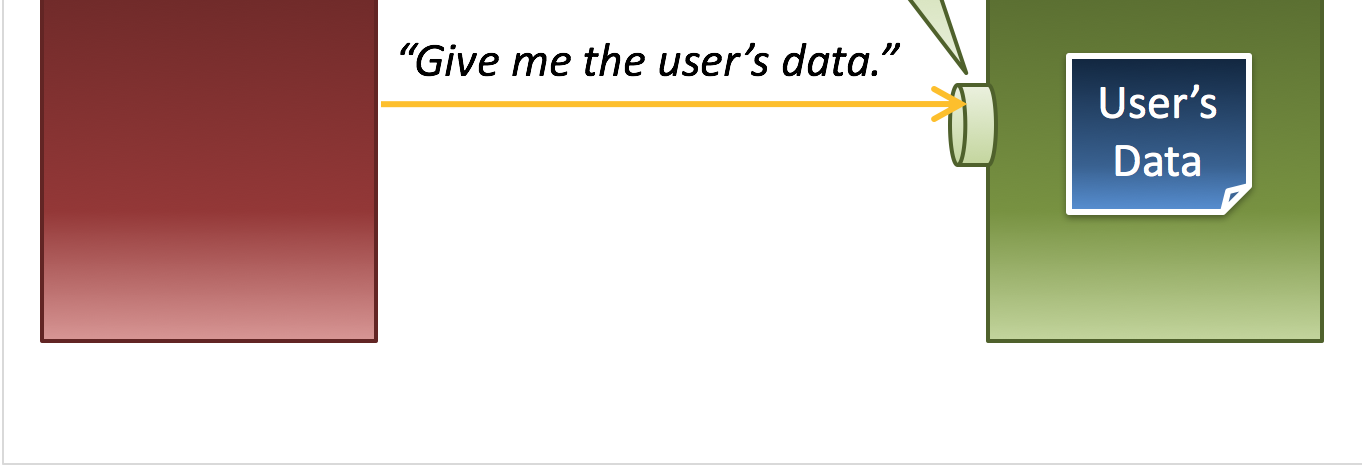


7. What if there is a malicious client application?

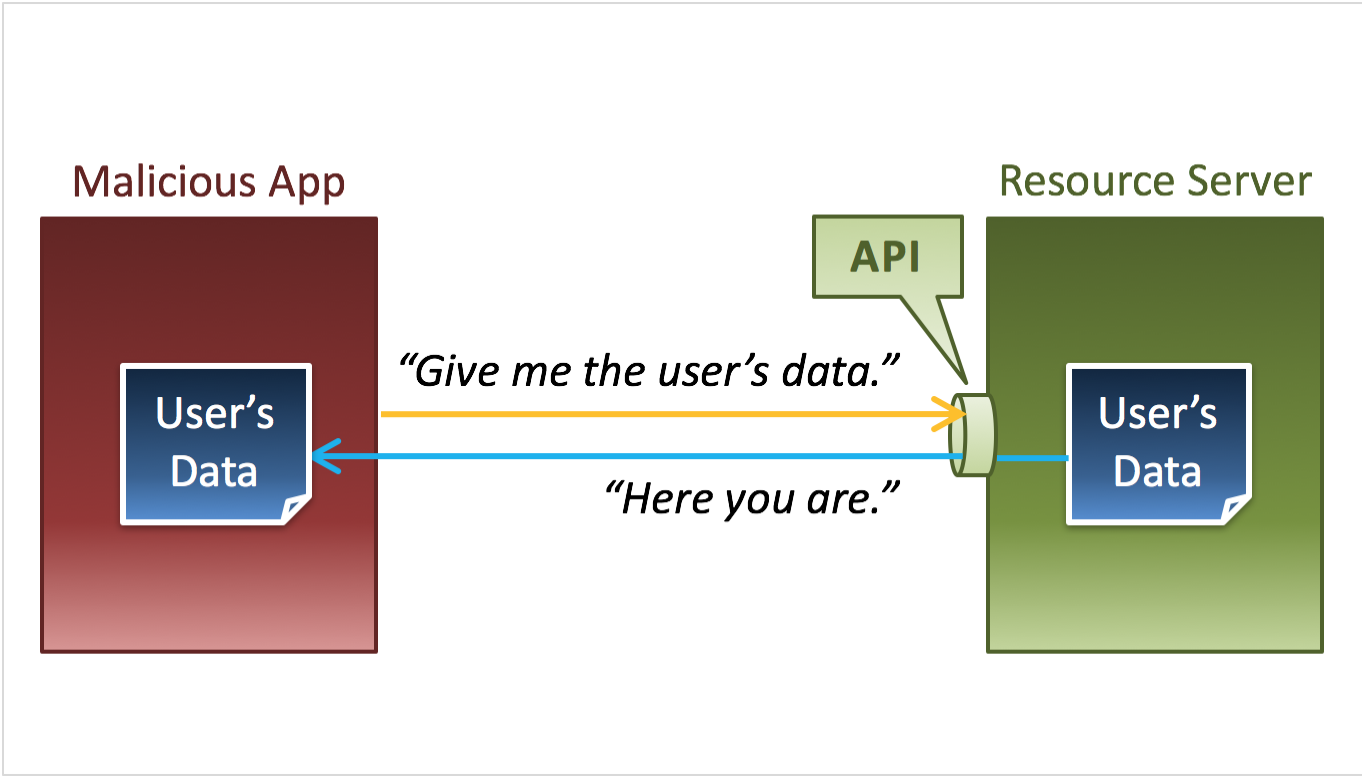


8. Even if the client application that requests the user's data is a malicious one, ...

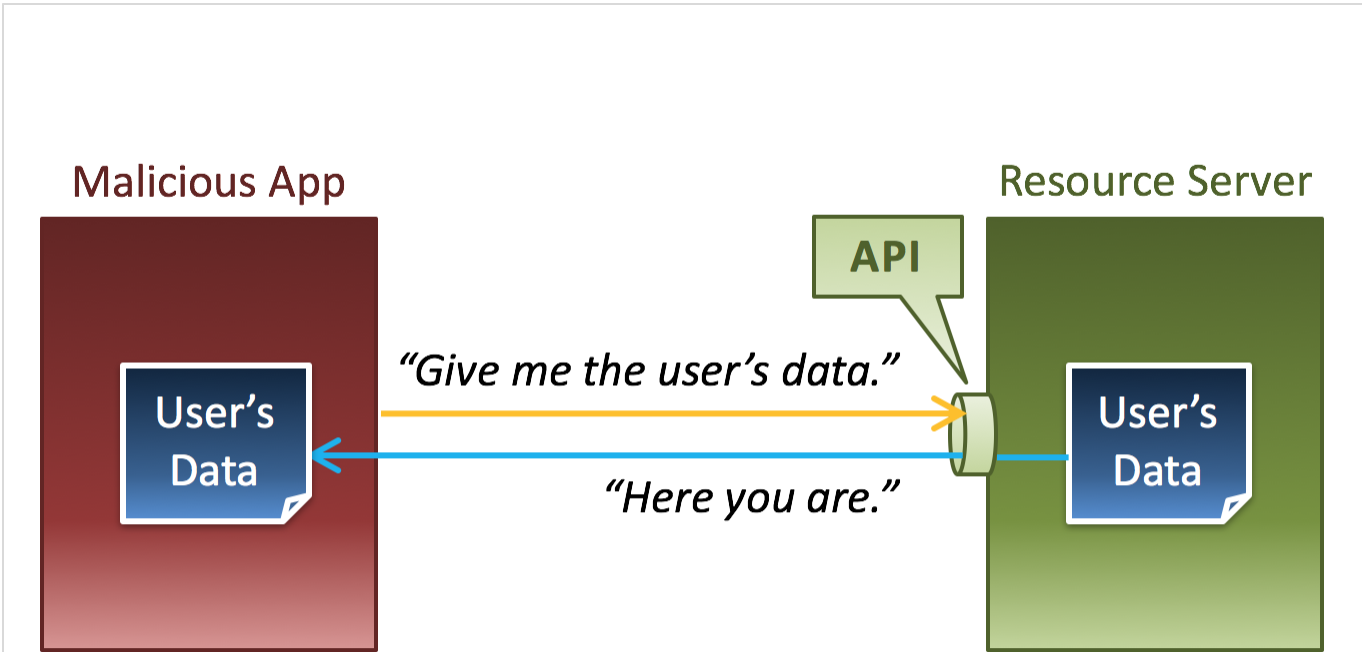




9. ... the resource server returns the user's data.

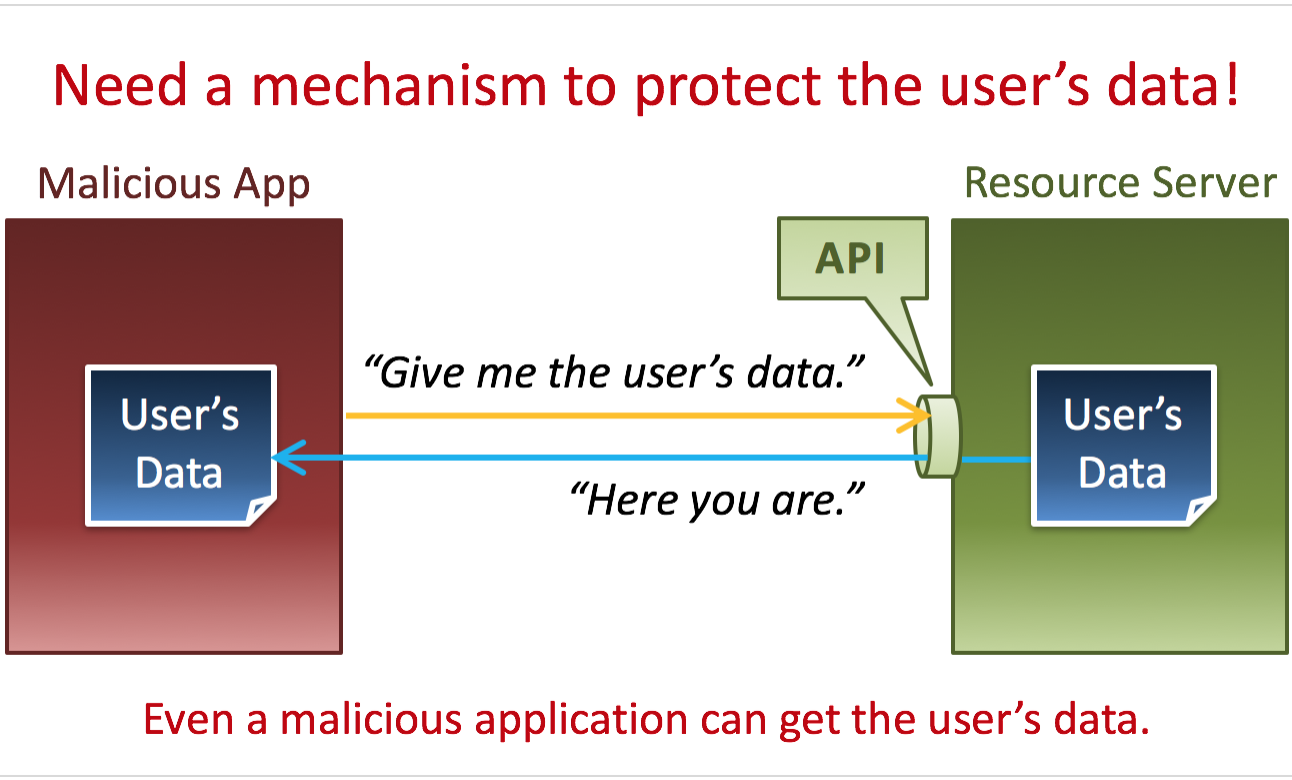


10. Even a malicious application can get the user's data.

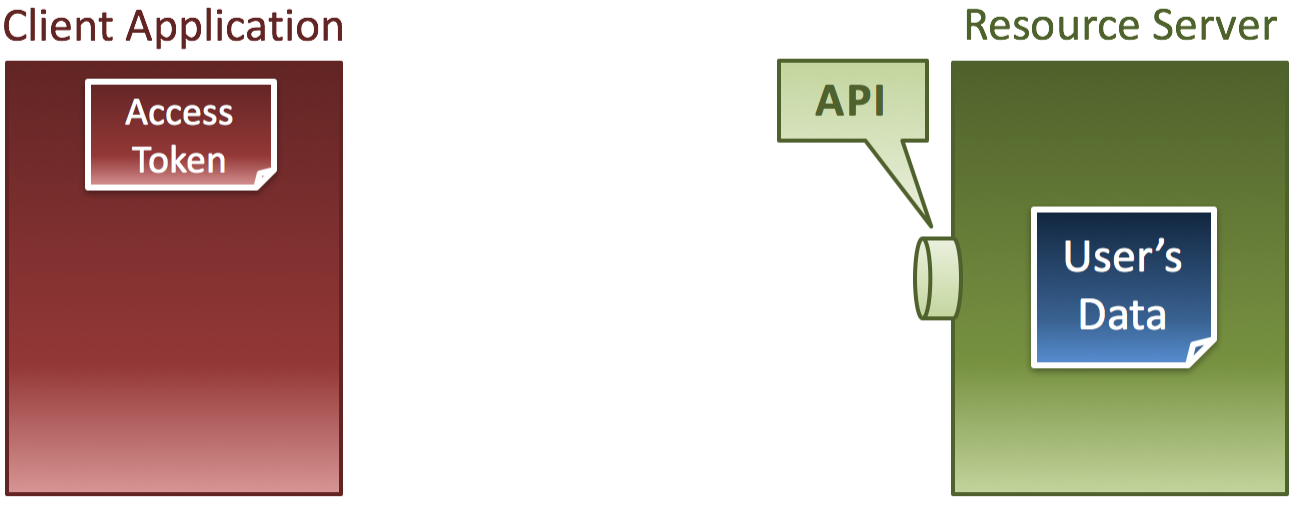


Even a malicious application can get the user's data.

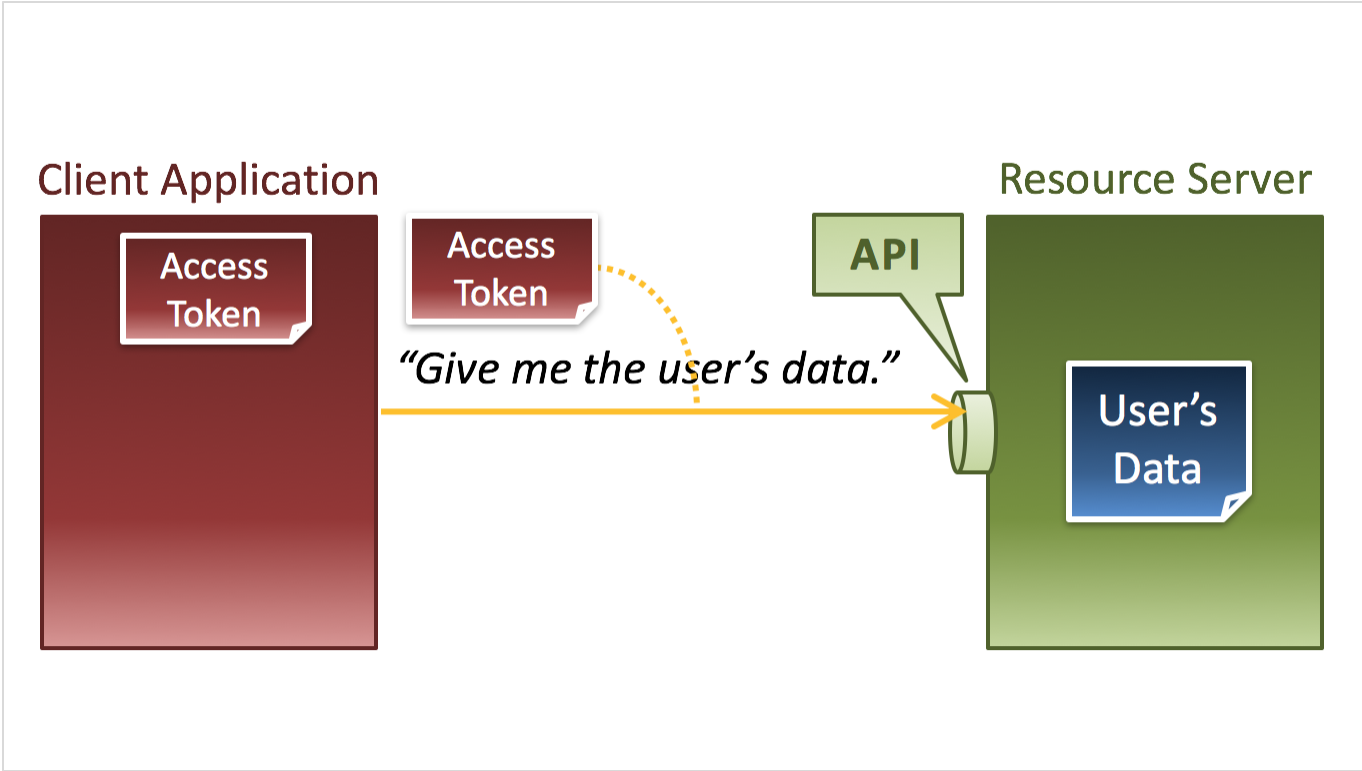
11. We need a mechanism to protect the user's data.



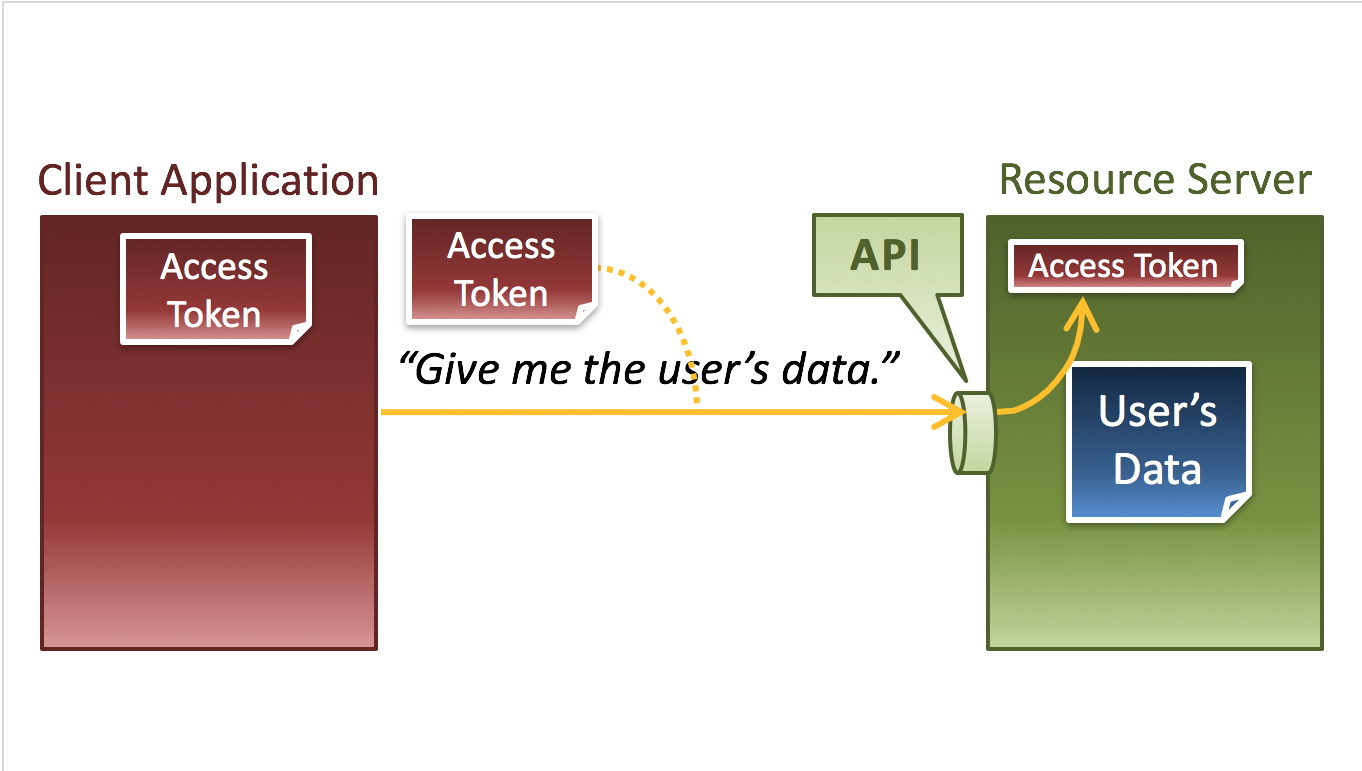
12. In the best practice, an "Access Token" is given to the client application in advance. An access token represents that the said client application has been given permissions to access the user's data.



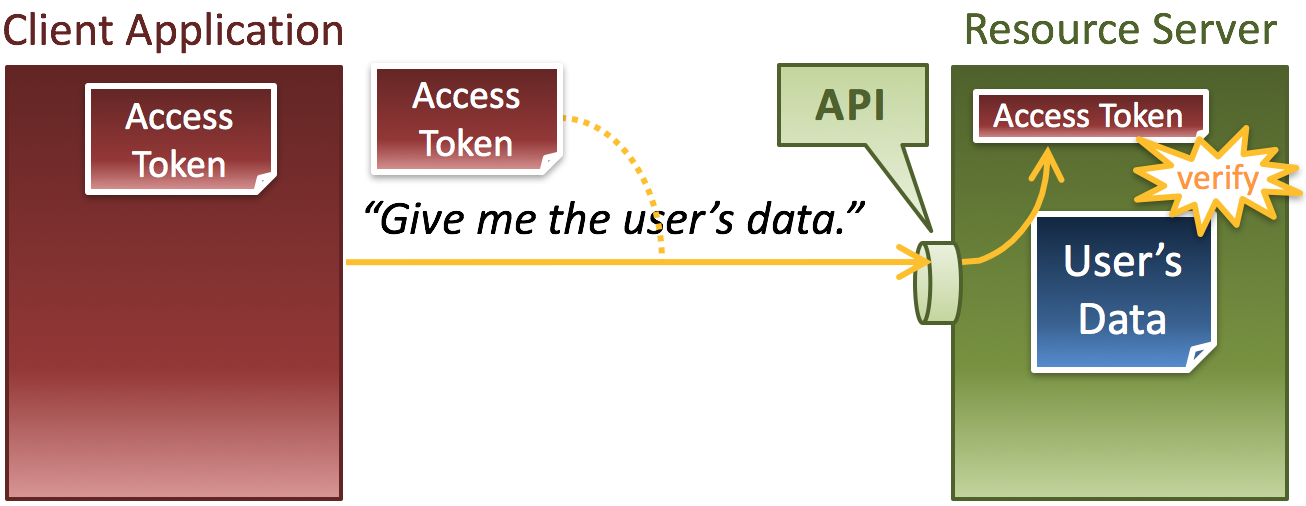
13. The client application presents the access token when it requests the user's data.



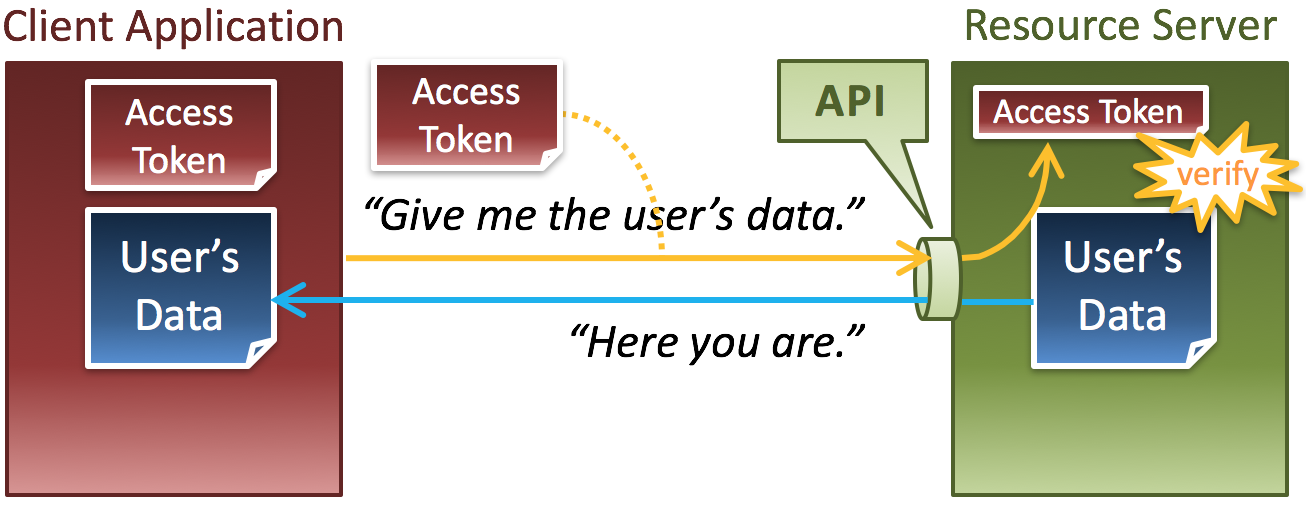
14. The resource server extracts the access token that is included in the request, ...



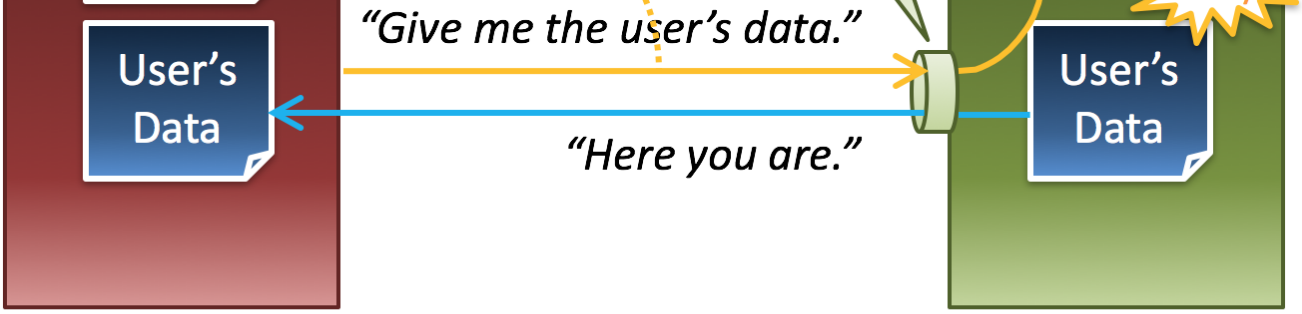
15. ... and confirms that the access token denotes that the client application has permissions to access the user's data.



16. After the confirmation, the resource server returns the user's data.



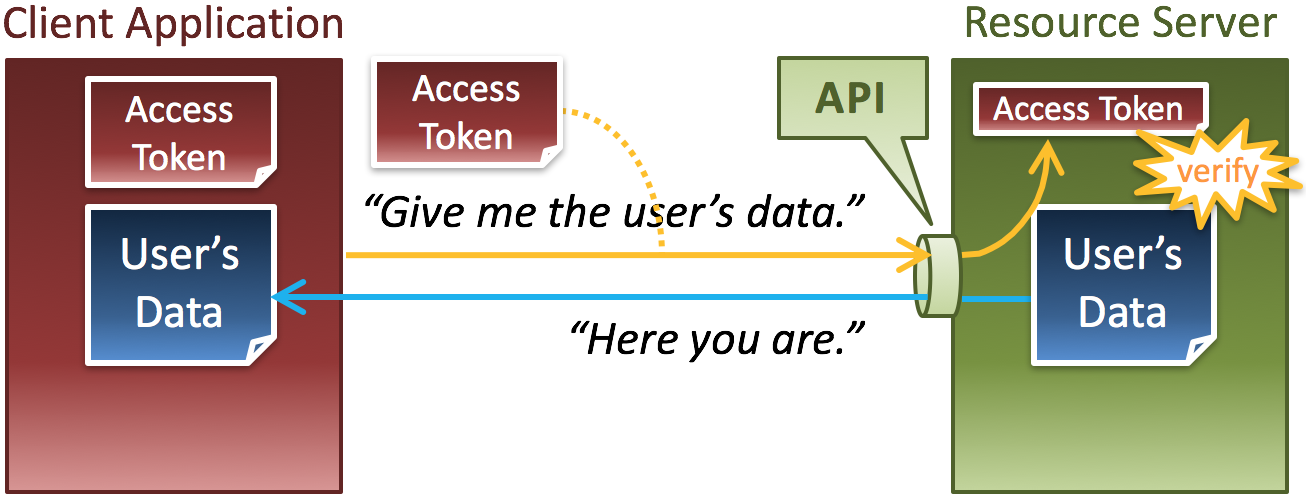
17. To make this mechanism work, an access token must be given to the client application in advance.



An access token must be given to the client in advance.

18. Consequently, we need someone who issues access tokens.

Need someone who issues access tokens!



An access token must be given to the client in advance.

19. Someone who issues access tokens ...

Someone who issues access tokens

20. ... is called "Authorization Server".

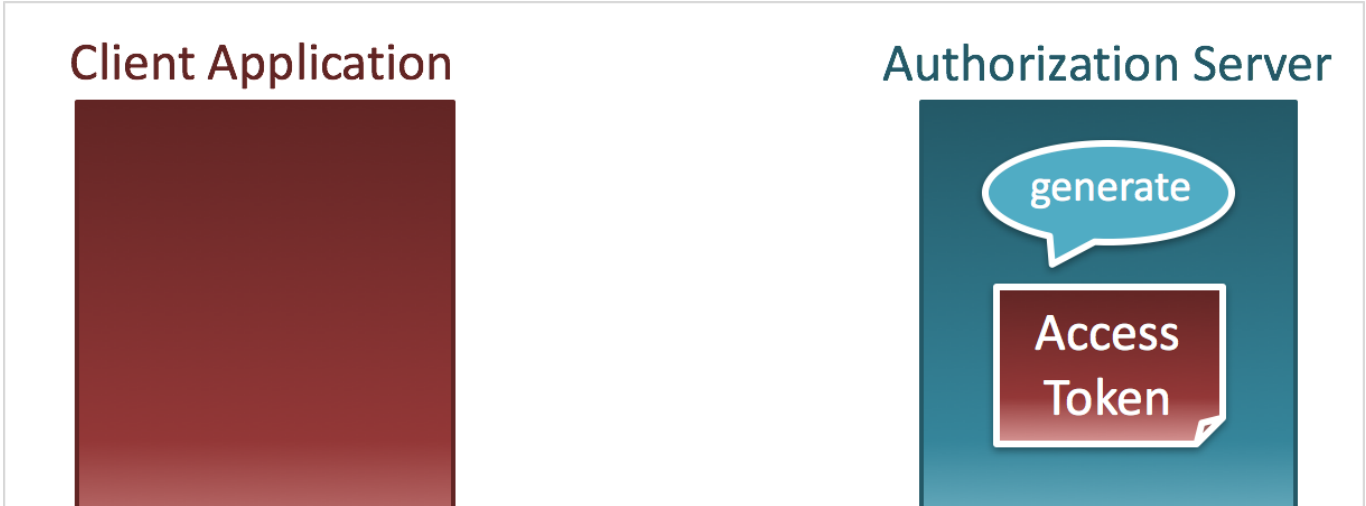
Someone who issues access tokens

||

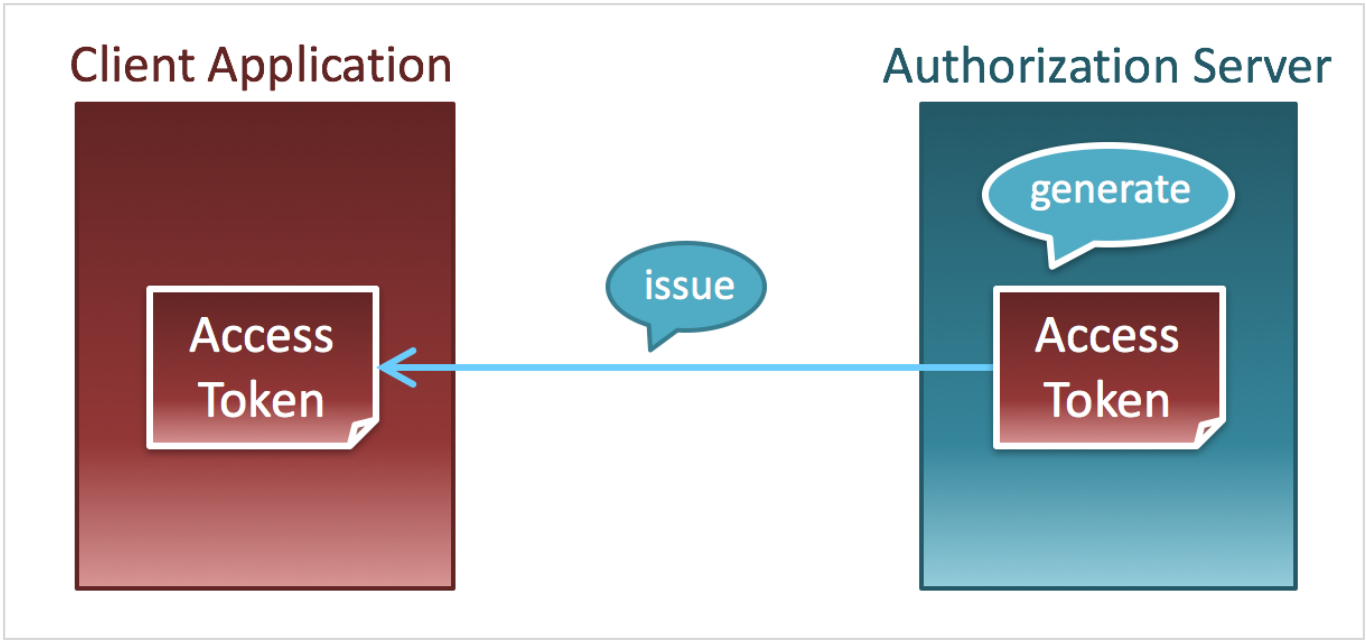
Authorization Server

21. The relationship between a client application and an authorization server is as follows.

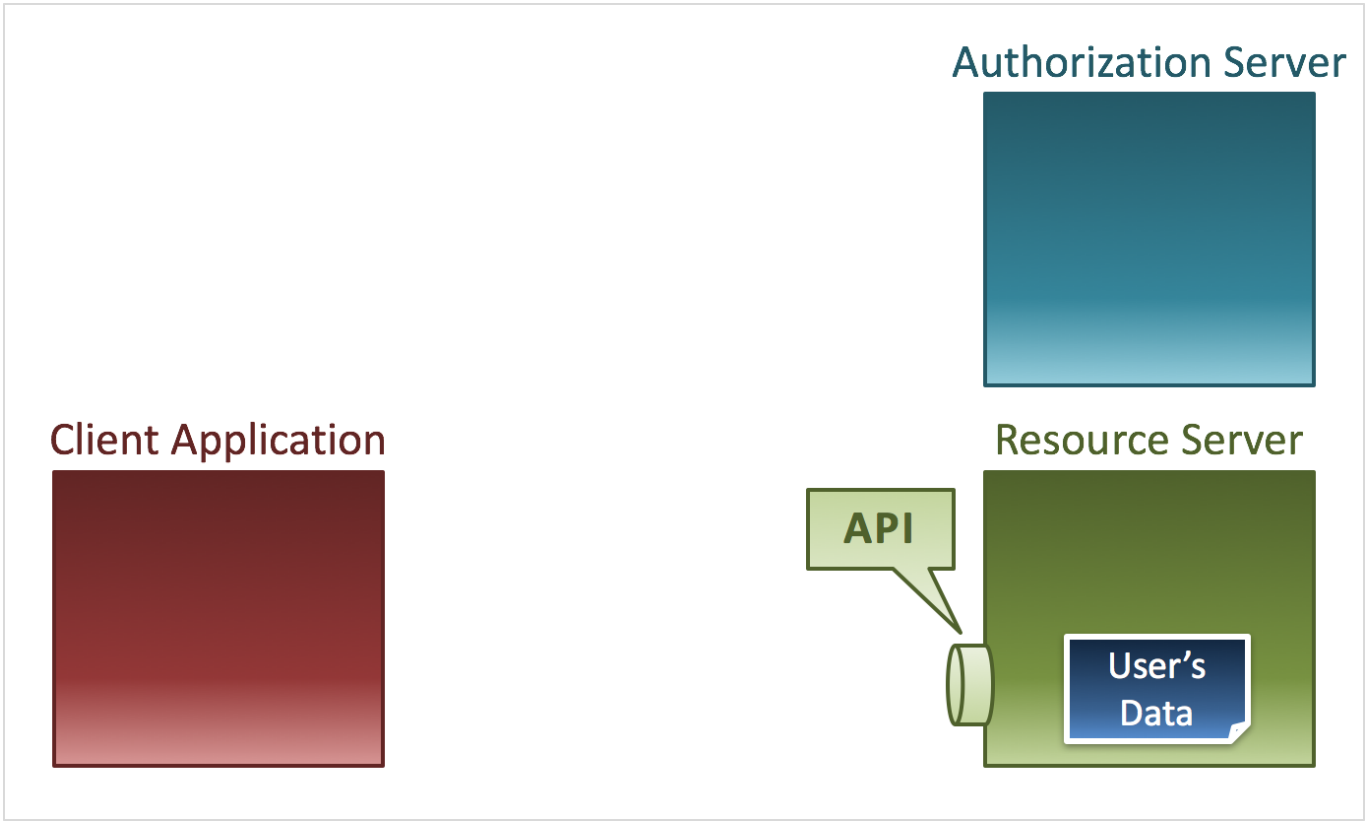
22. An authorization server generates an access token ...



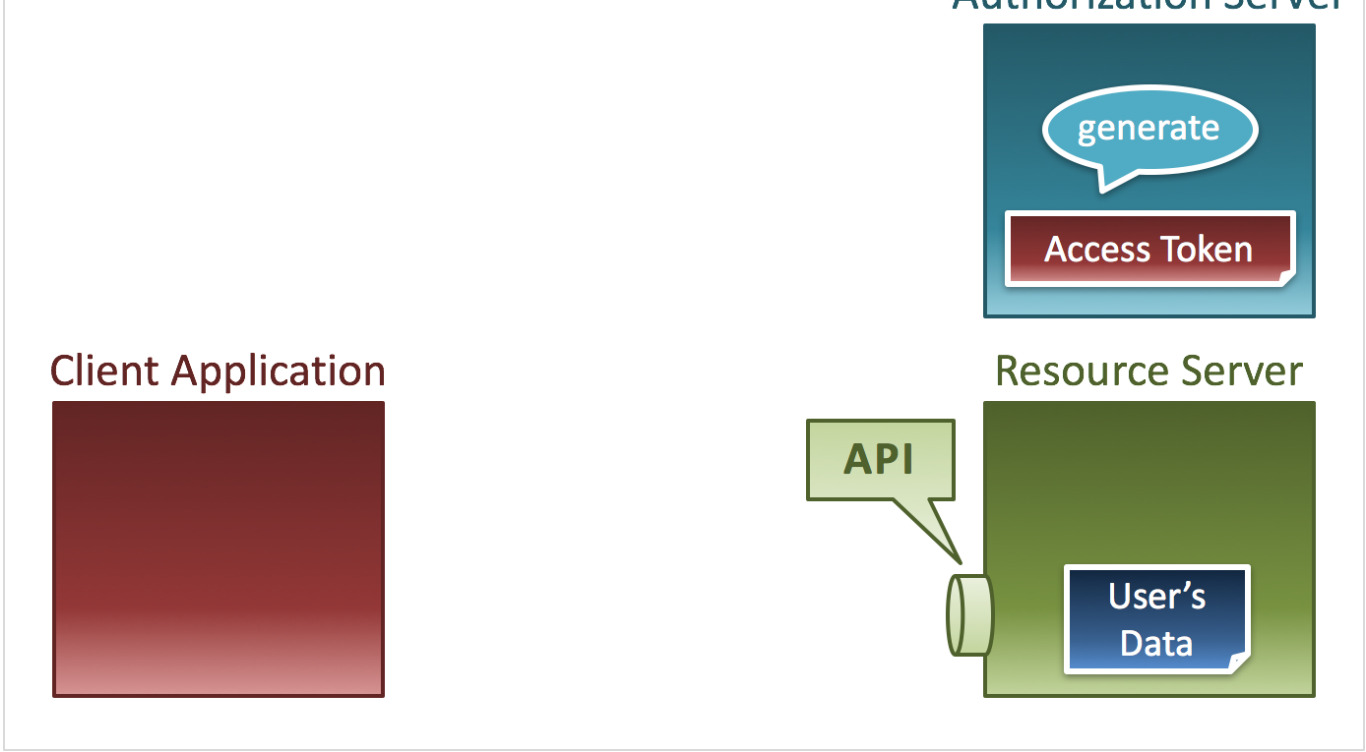
23. ... and issues the access token to a client application.



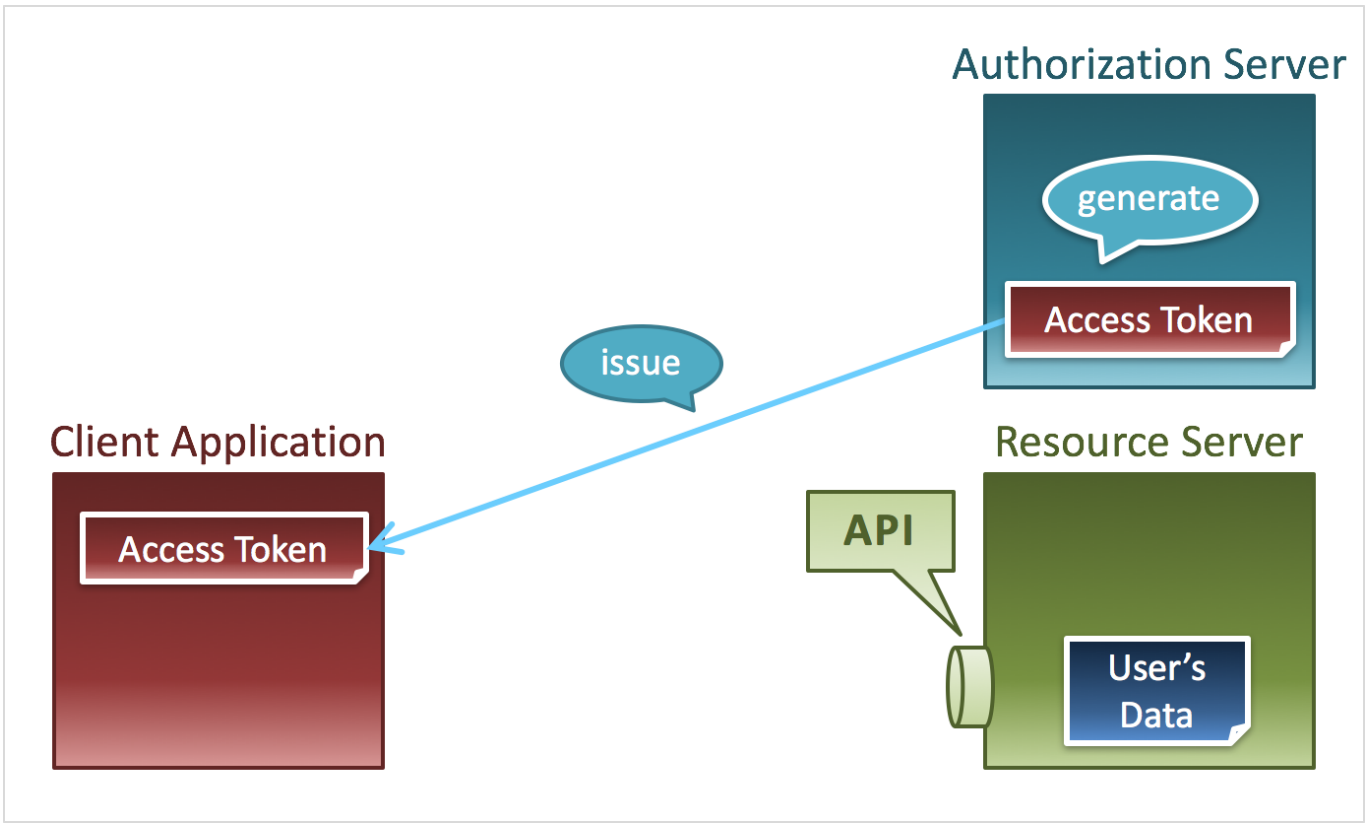
24. Let's review what we've learned so far. Characters are an "Authorization Server", a "Client Application" and a "Resource Server".



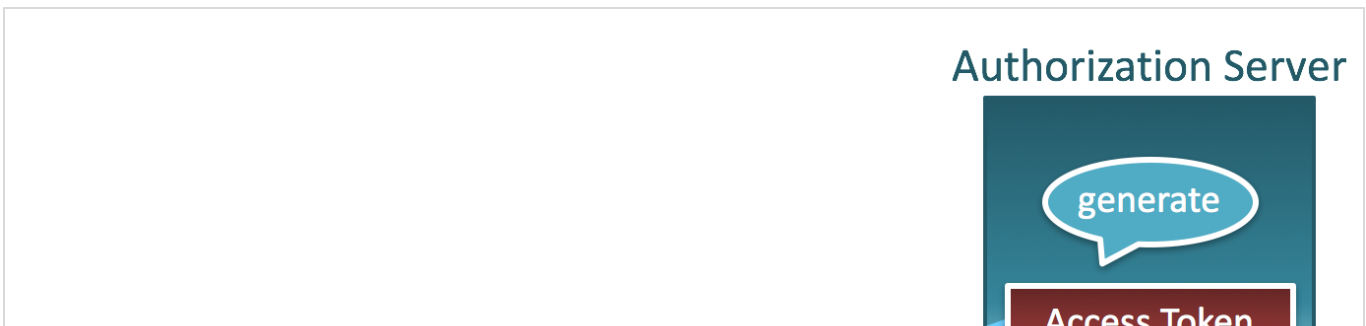
25. The authorization server generates an access token ...

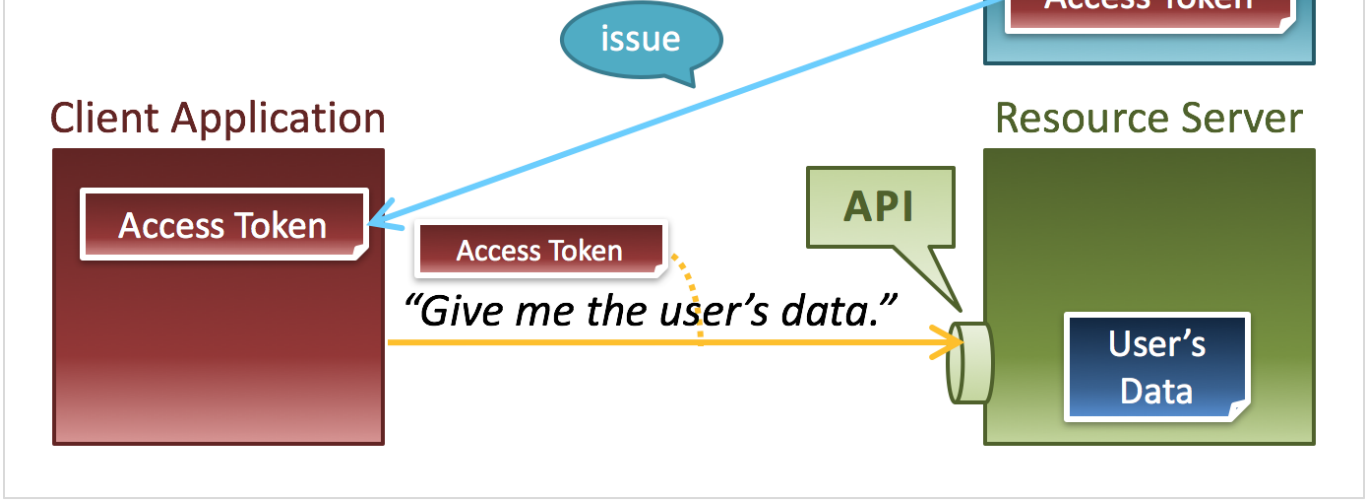


26. ... and issues the access token to the client application.

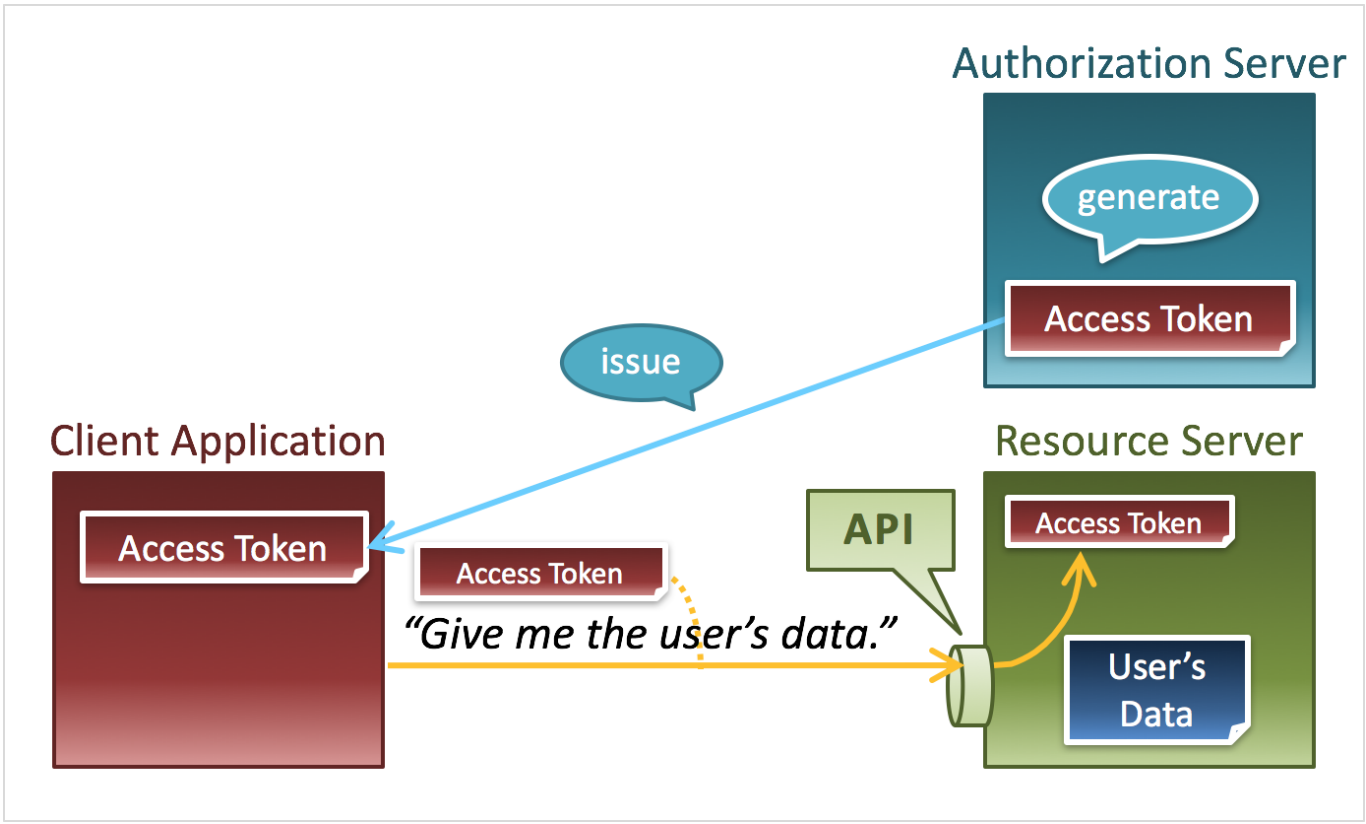


27. The client application requests the user's data with the access token.

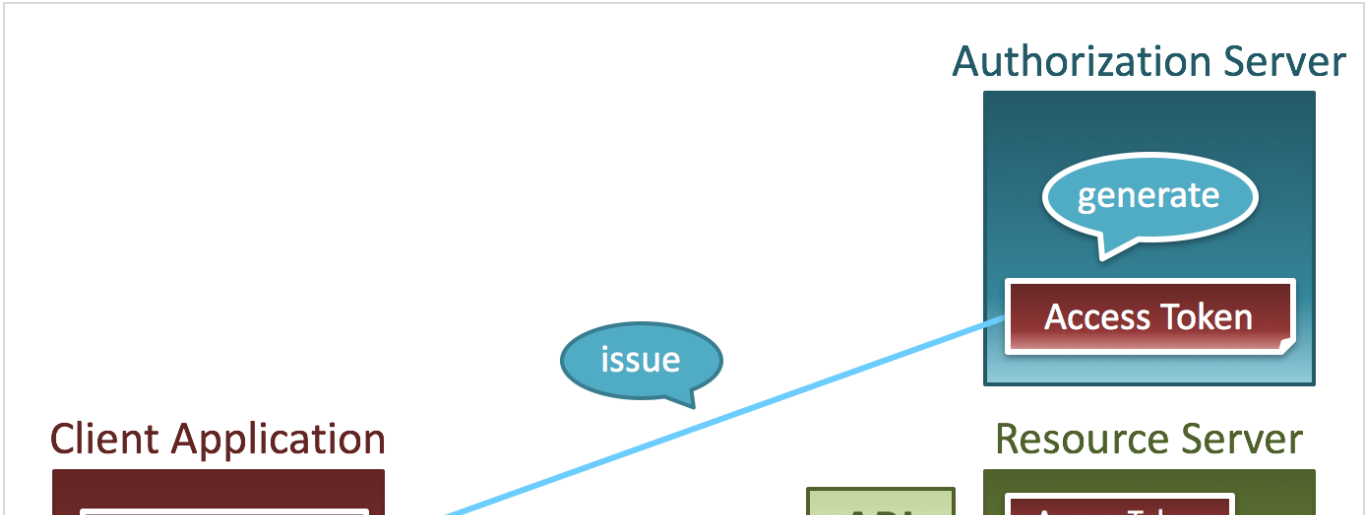




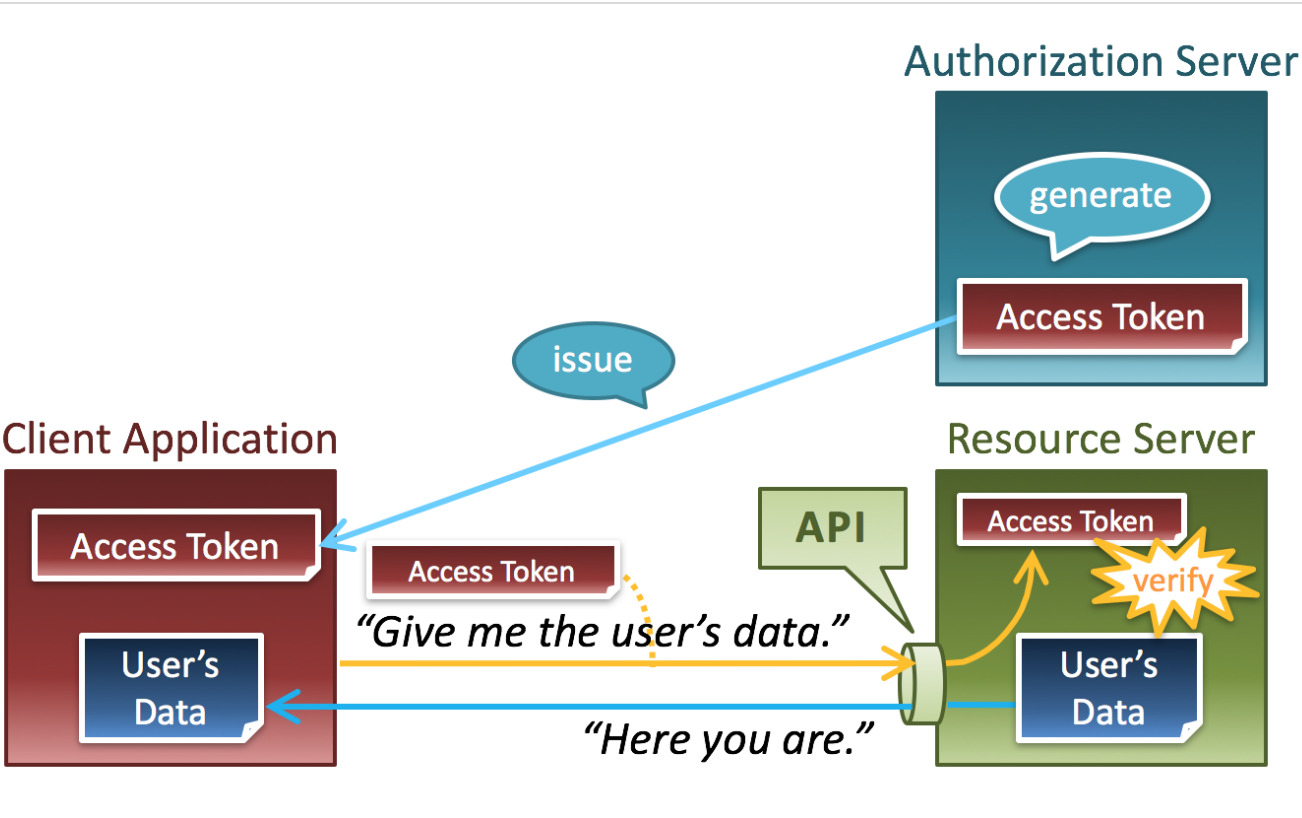
28. The resource server extracts the access token from the request, ...



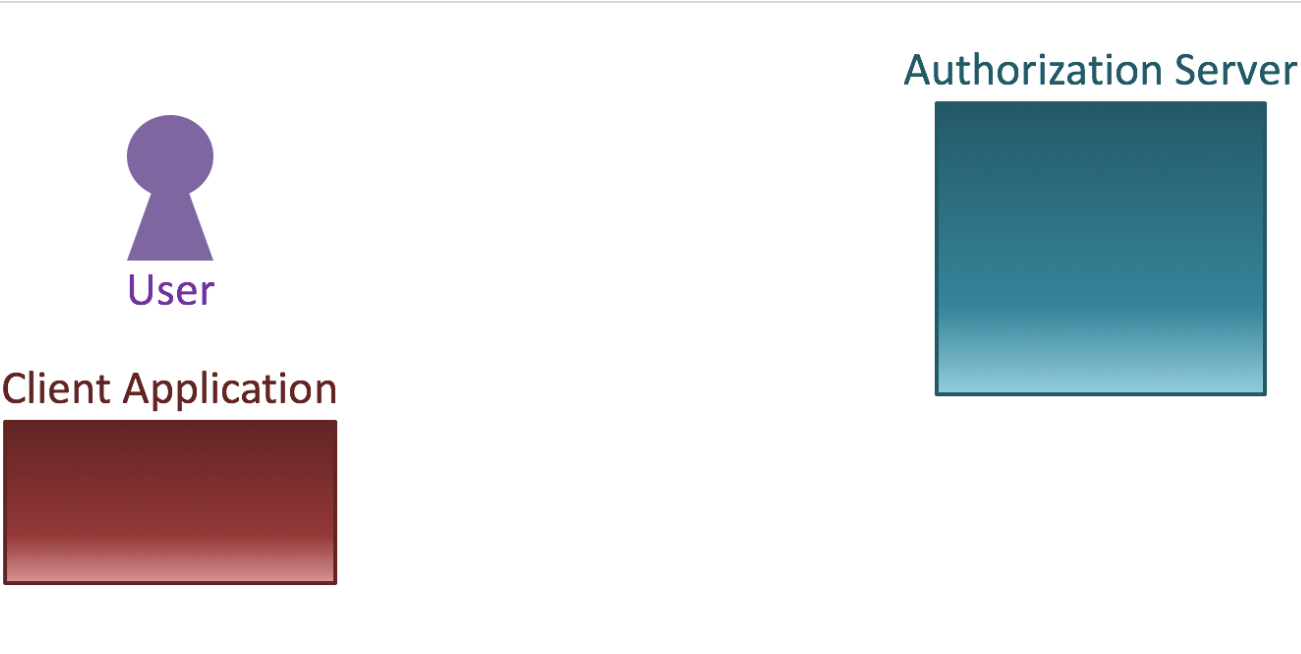
29. ... confirms that the access token has permissions to access the user's data ...



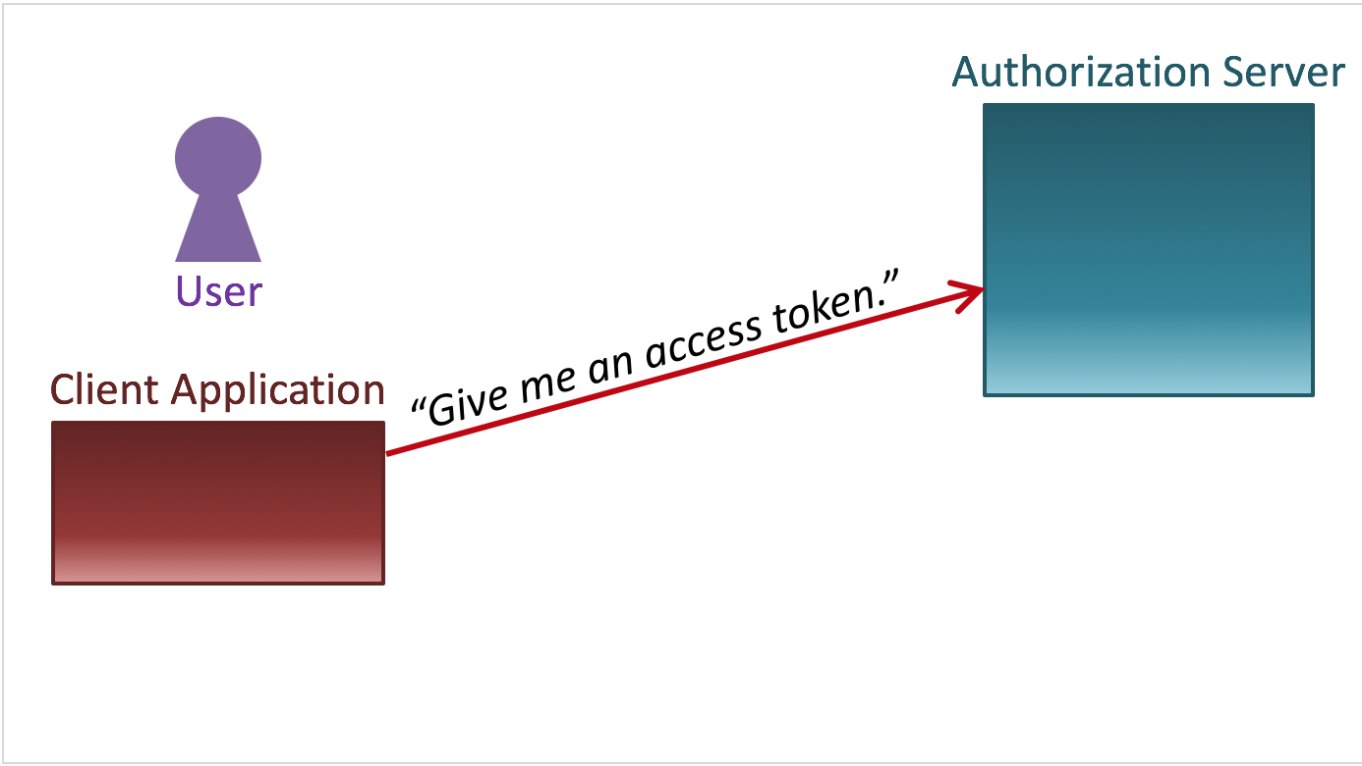
30. ... and returns the user's data to the client application.



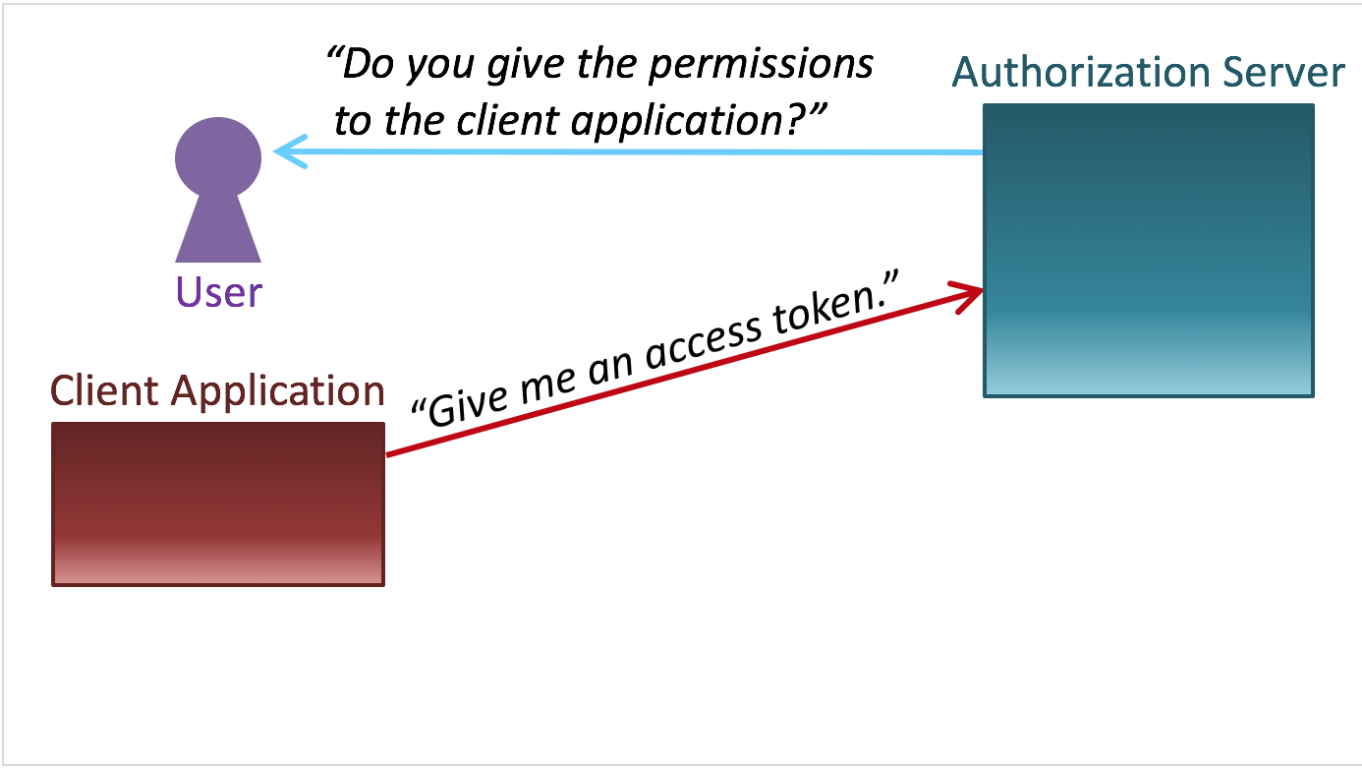
31. In the flow above, the first step is access token generation by an authorization server. However, in a real flow, the user is asked before an access token is issued.



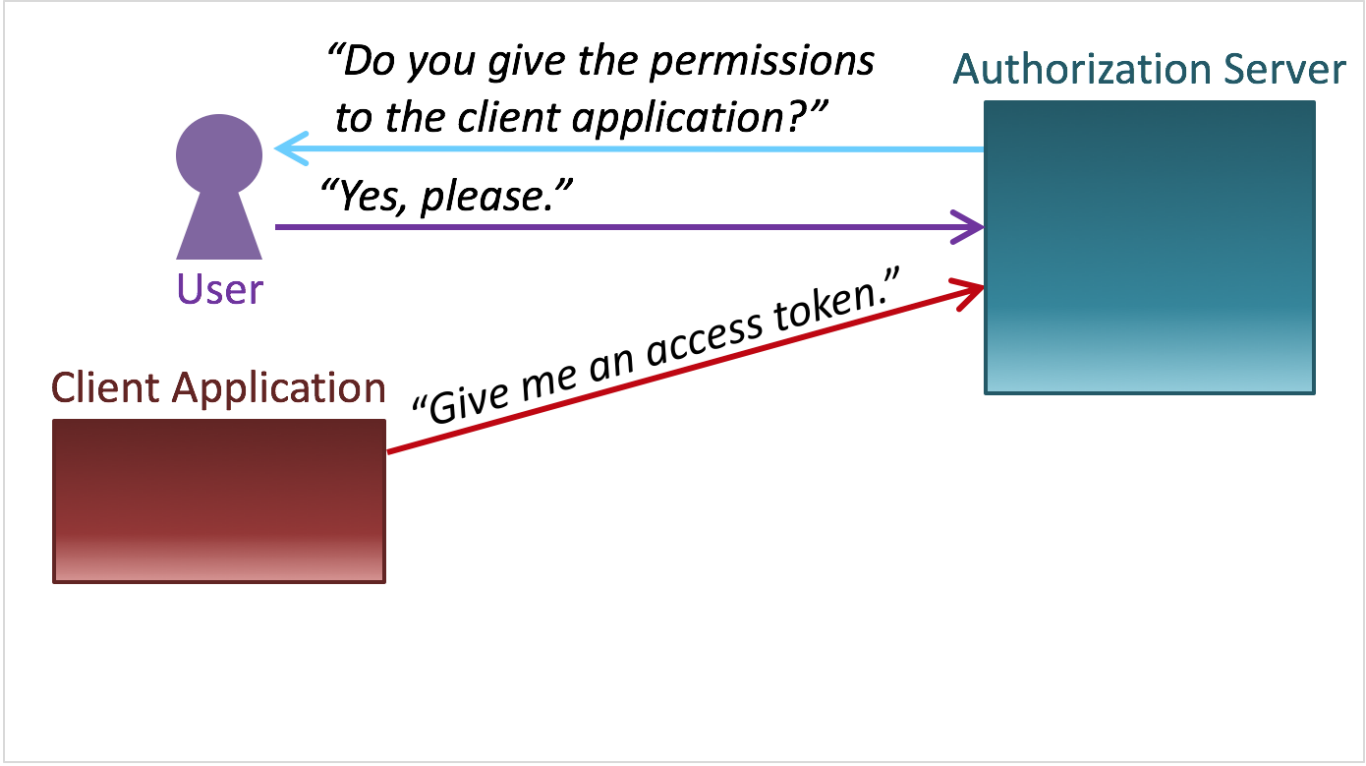
32. First, the client application requests an access token.



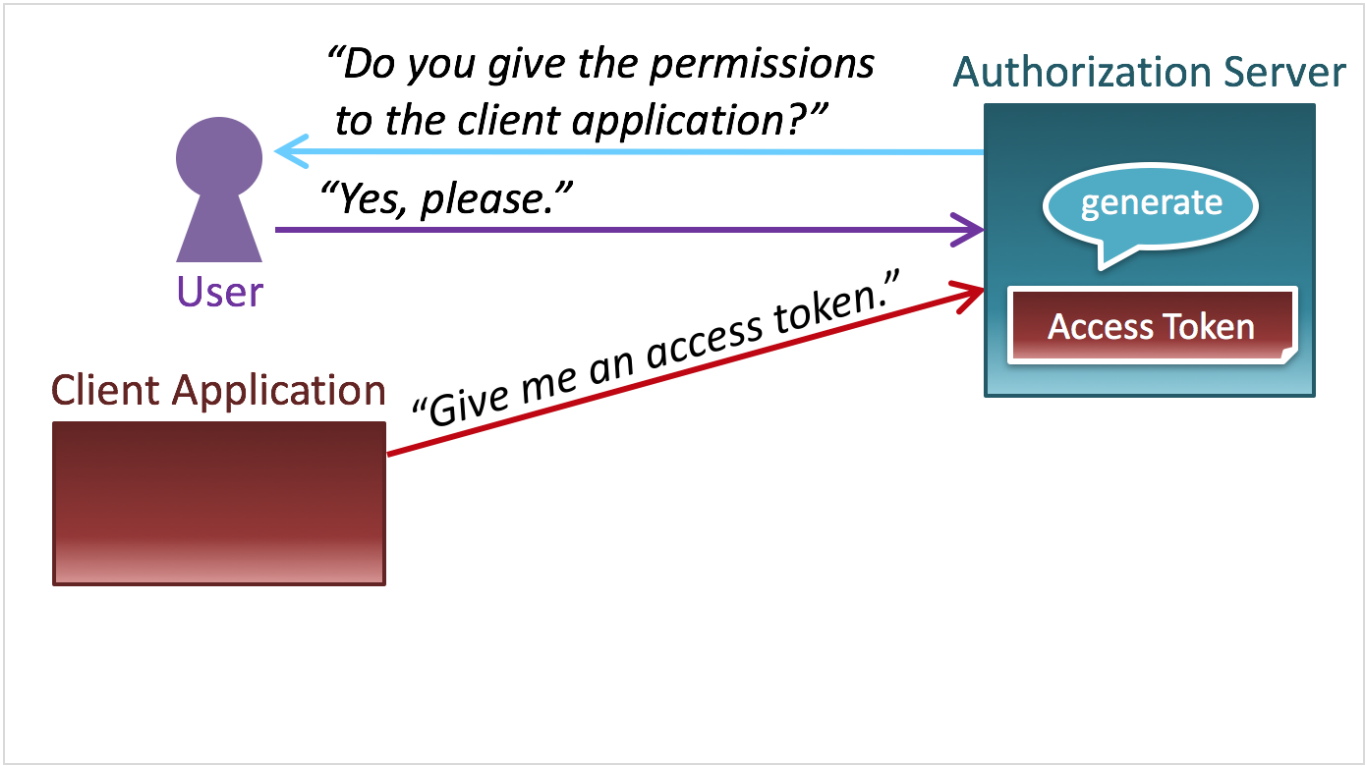
33. Then, the authorization server asks the user whether to grant the requested permissions to the client application.



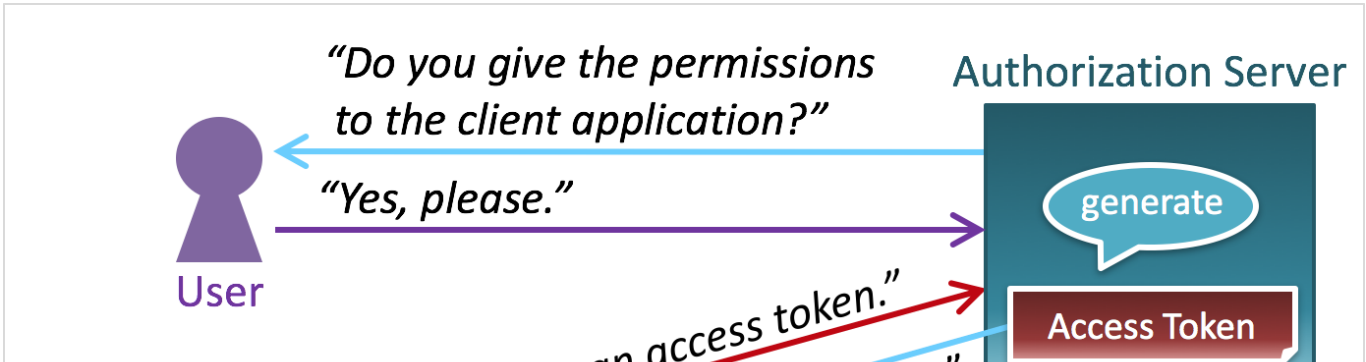
34. If the user allows the authorization server to issue an access token to the client application, ...

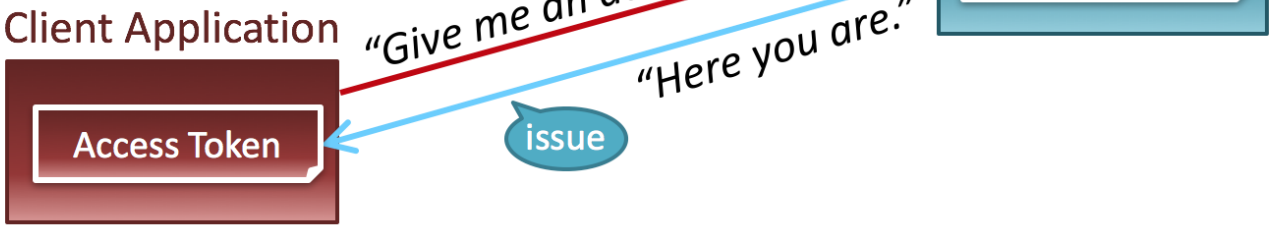


35. ... the authorization server generates an access token ...

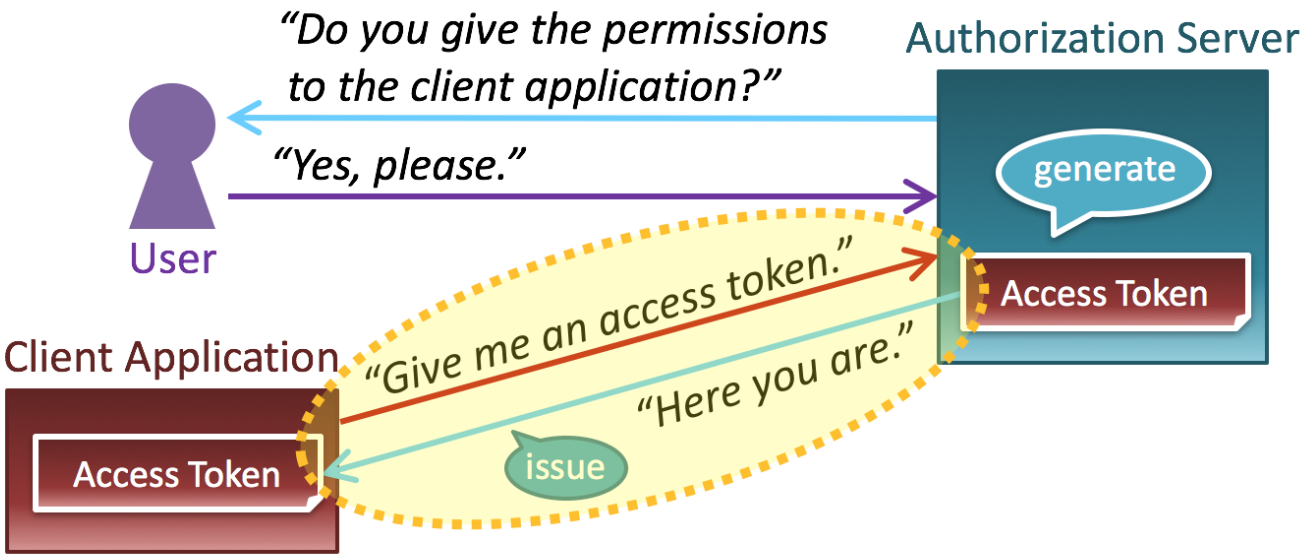


36. ... and issues the access token to the client application.

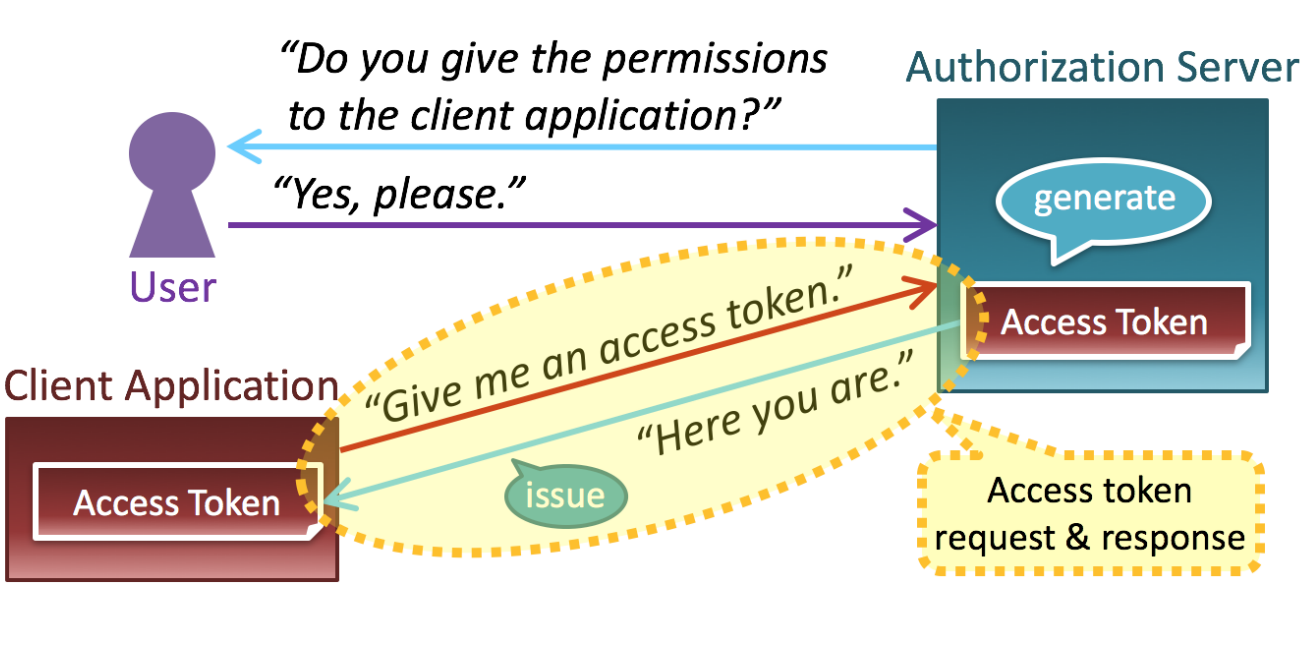




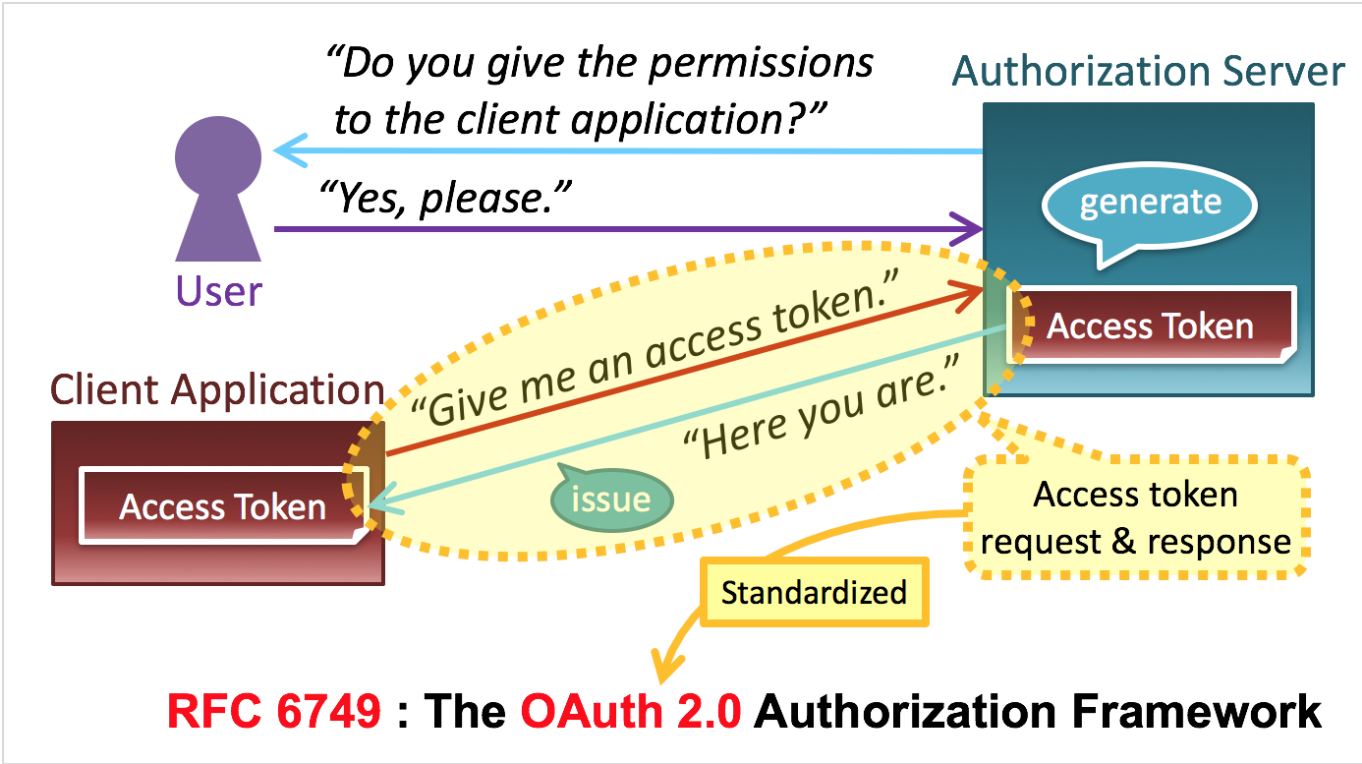
37. By the way, pay attention to the part encircled by the yellow ellipse.



38. The part represents an access token request and a response to the request.



39. And, it is "OAuth 2.0" that has standardized the part. Details of OAuth 2.0 are described in the technical document, RFC 6749 (The OAuth 2.0 Authorization Framework).



Next To Read

Diagrams And Movies Of All The OAuth 2.0 Flows

Oauth