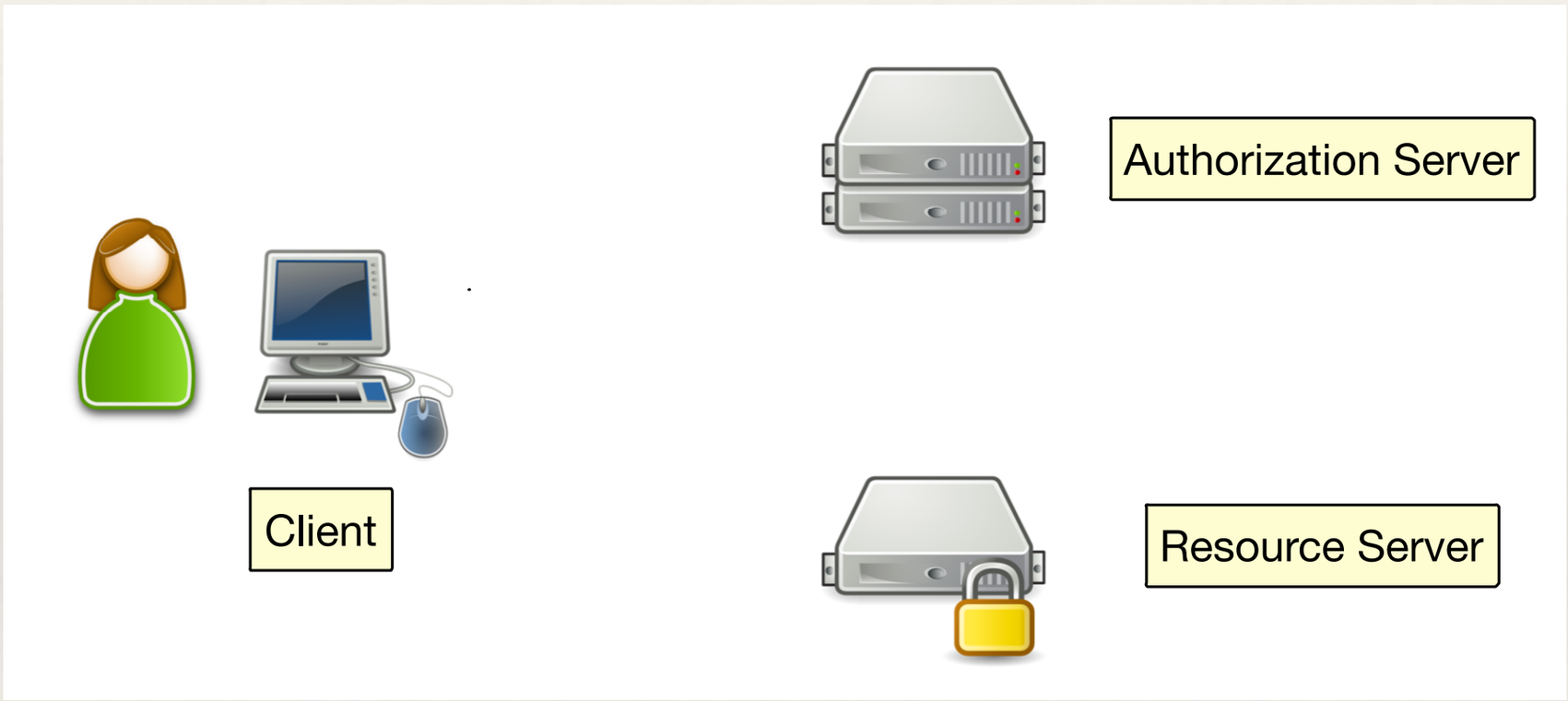
OAuth2

RFC6749
RFC6750

- The OAuth 2.0 authorization framework enables a third-party application to obtain limited access to an HTTP service,
  - either on behalf of a resource owner by orchestrating an approval interaction between the resource owner and the HTTP service,
  - or by allowing the third-party application to obtain access on its own behalf.

The players

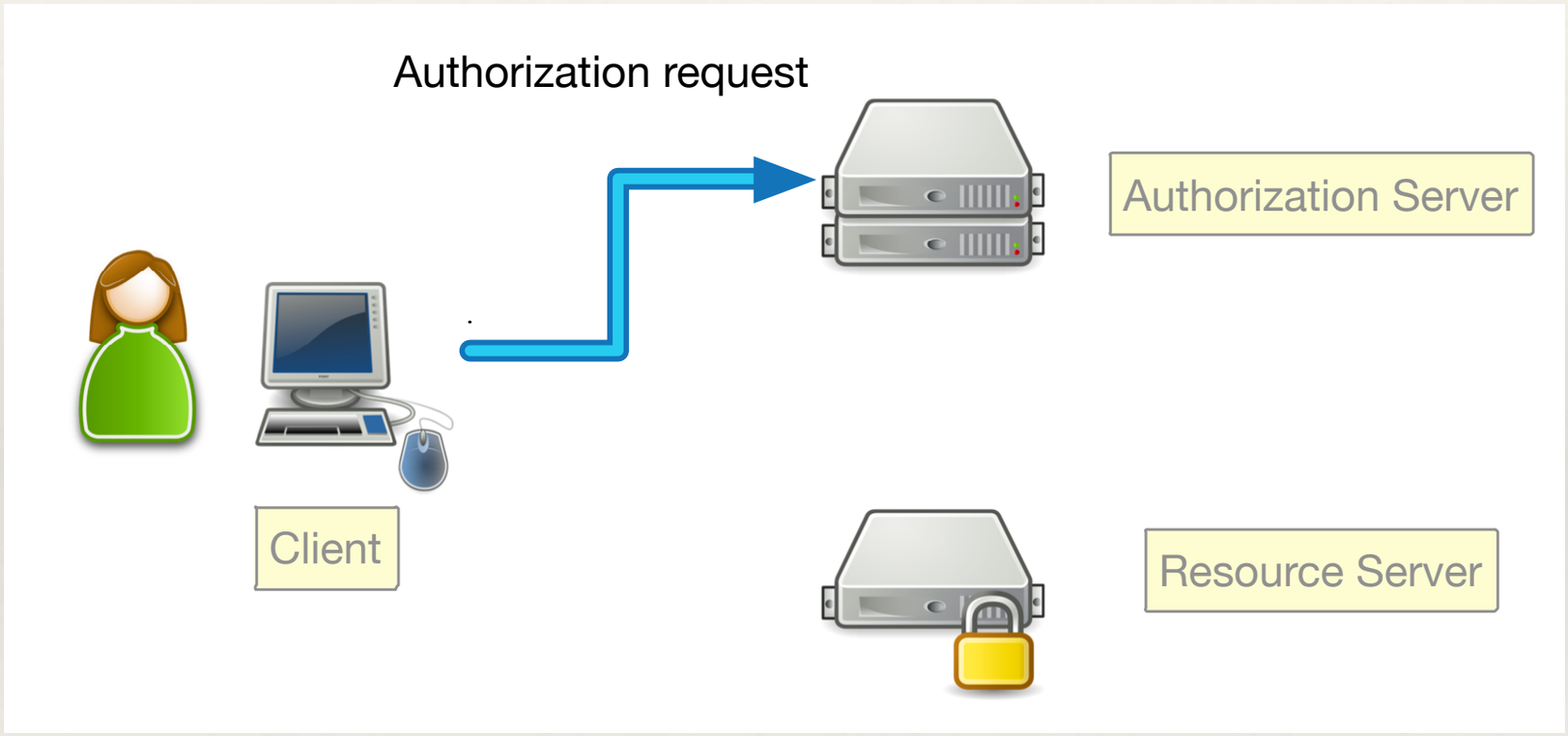


Flows

- Authorization Code Grant (Code)
- Implicit Grant (Implicit)
- Resource Owner Password Credentials Grant
- Client Credentials Grant

Code flow

Authorization Request



Authorization Request - details

Parameters

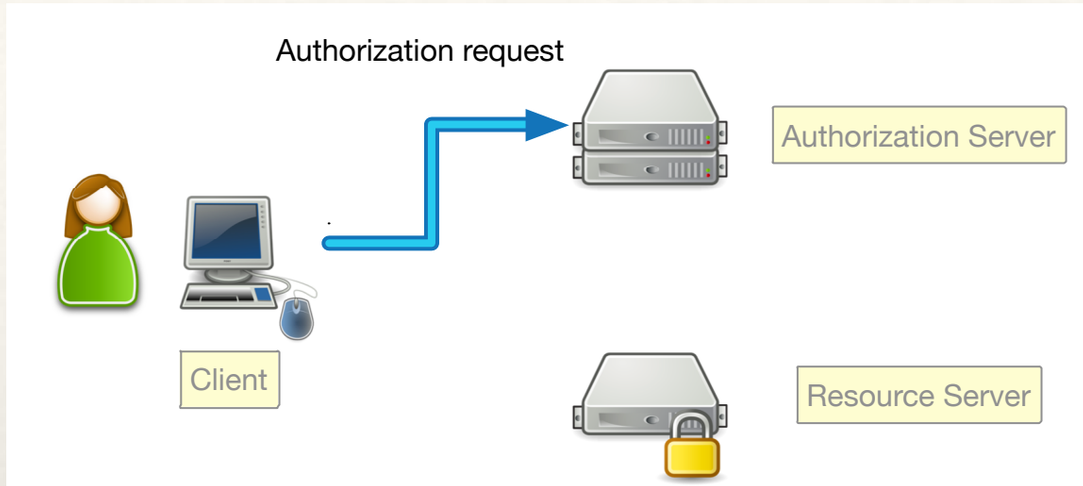
client_id

redirect_uri

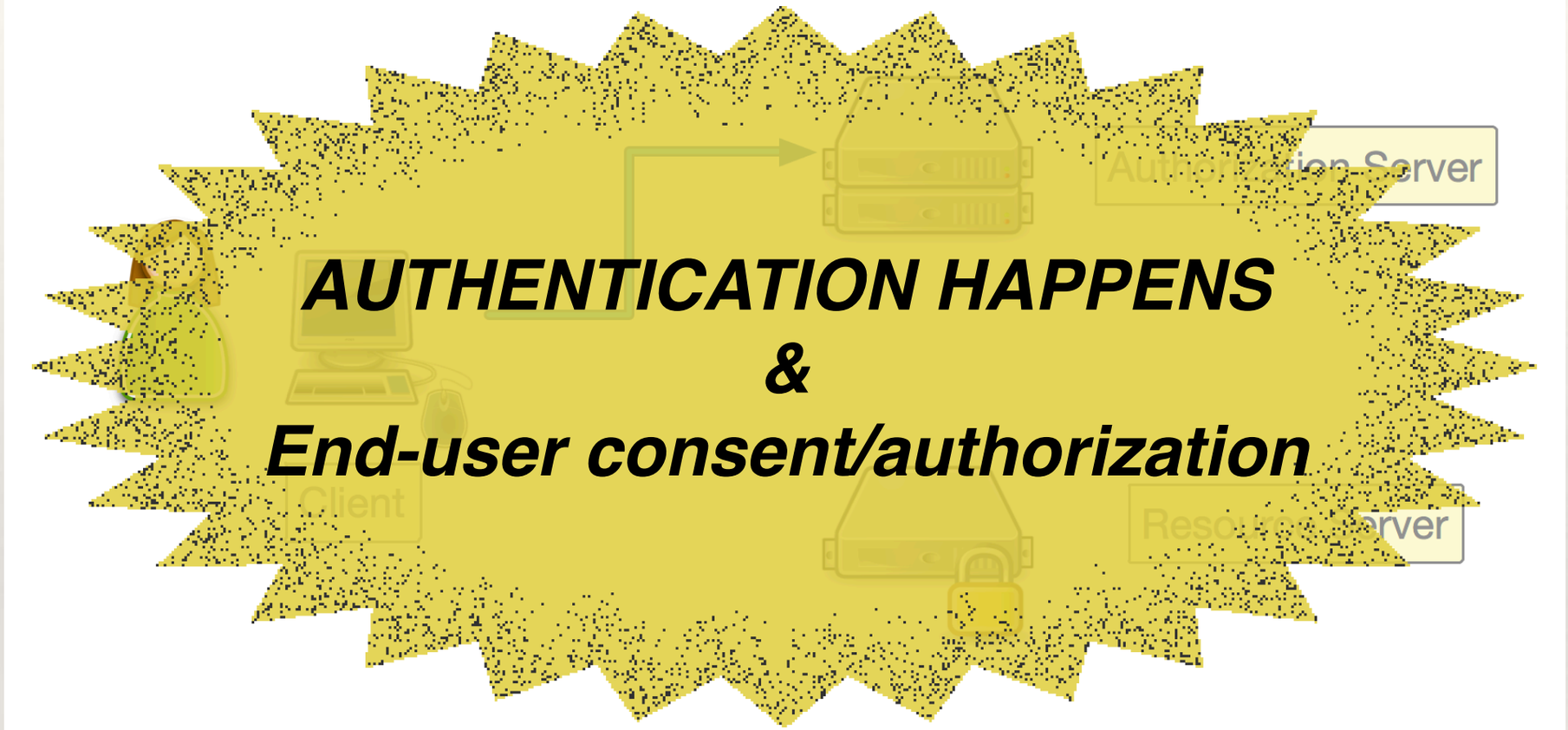
response_type

scope

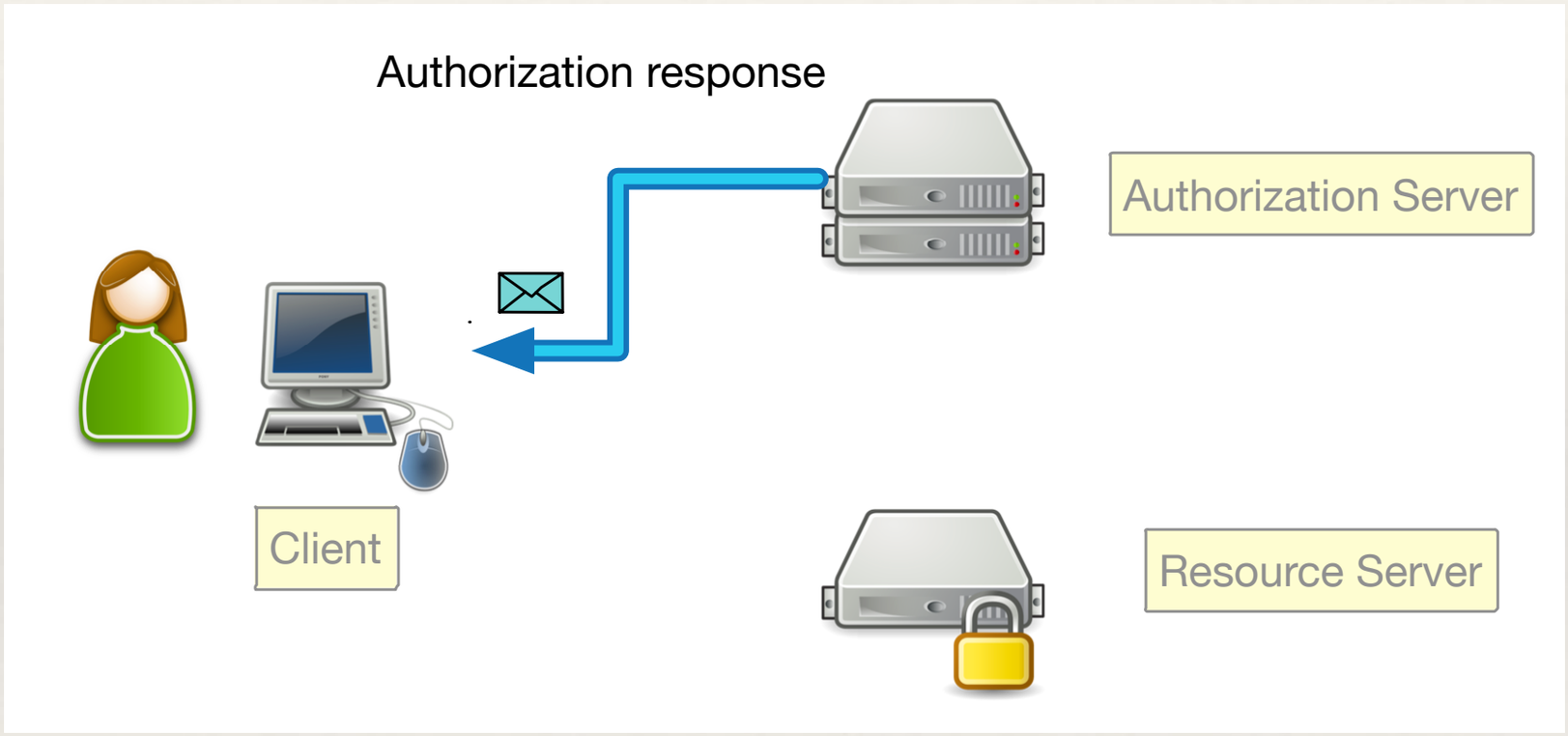
state




```
GET/authorization?state=15216719803168020 35&
      redirect_uri=https://example.org/authz_cb&
      response_type=code&
      client_id=SFEBuhC7sp3a
Host: as.example.com
```


AUTHENTICATION HAPPENS
&
End-user consent/authorization

Authorization Response

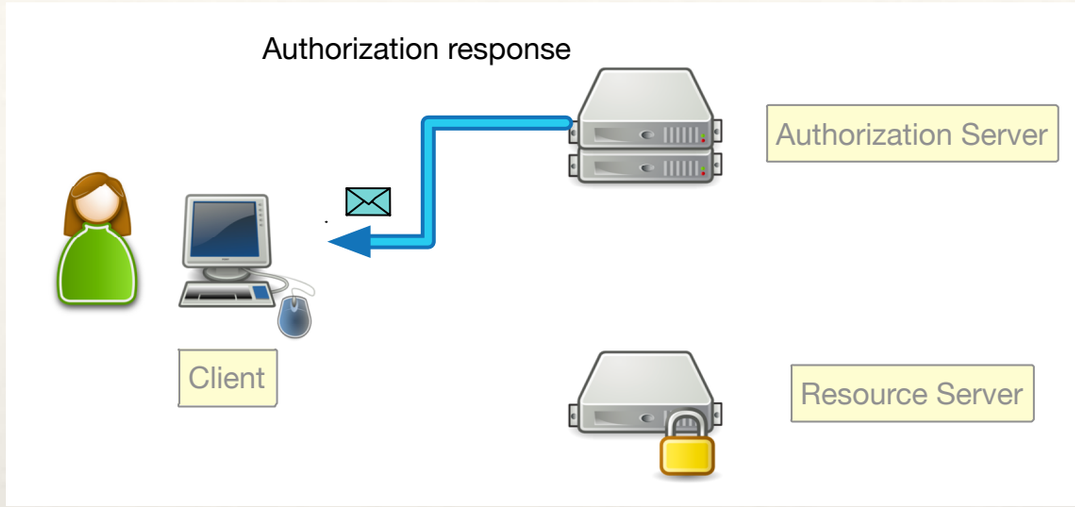


Authorization Response - details

Parameters

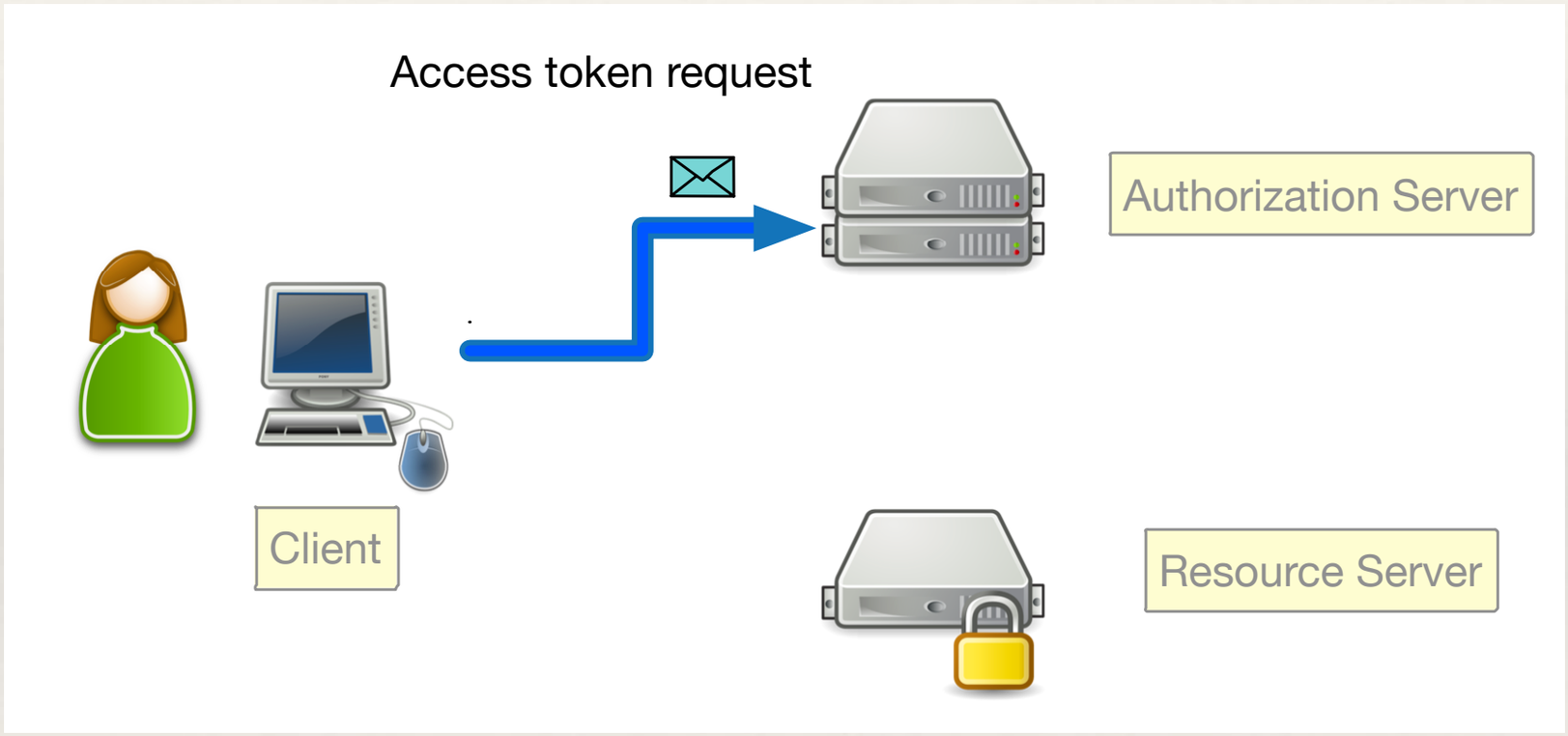
code

state



HTTP/1.1 302 Found
Location: https://rp.example.com/authz_cb?state=15216719803168020 35&
code=s87BT60pp2UbNX2HnkWpZ9YhPVHRZaoTuU9XJul6JMuQaKUidUM6y1Boab6

Access Token Request



Access Token Request - details

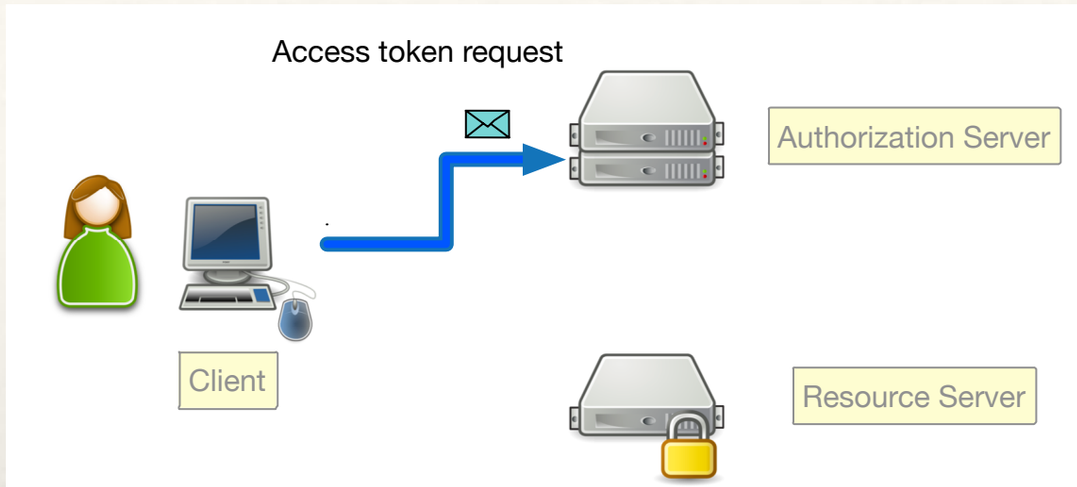
Parameters

client_id

code

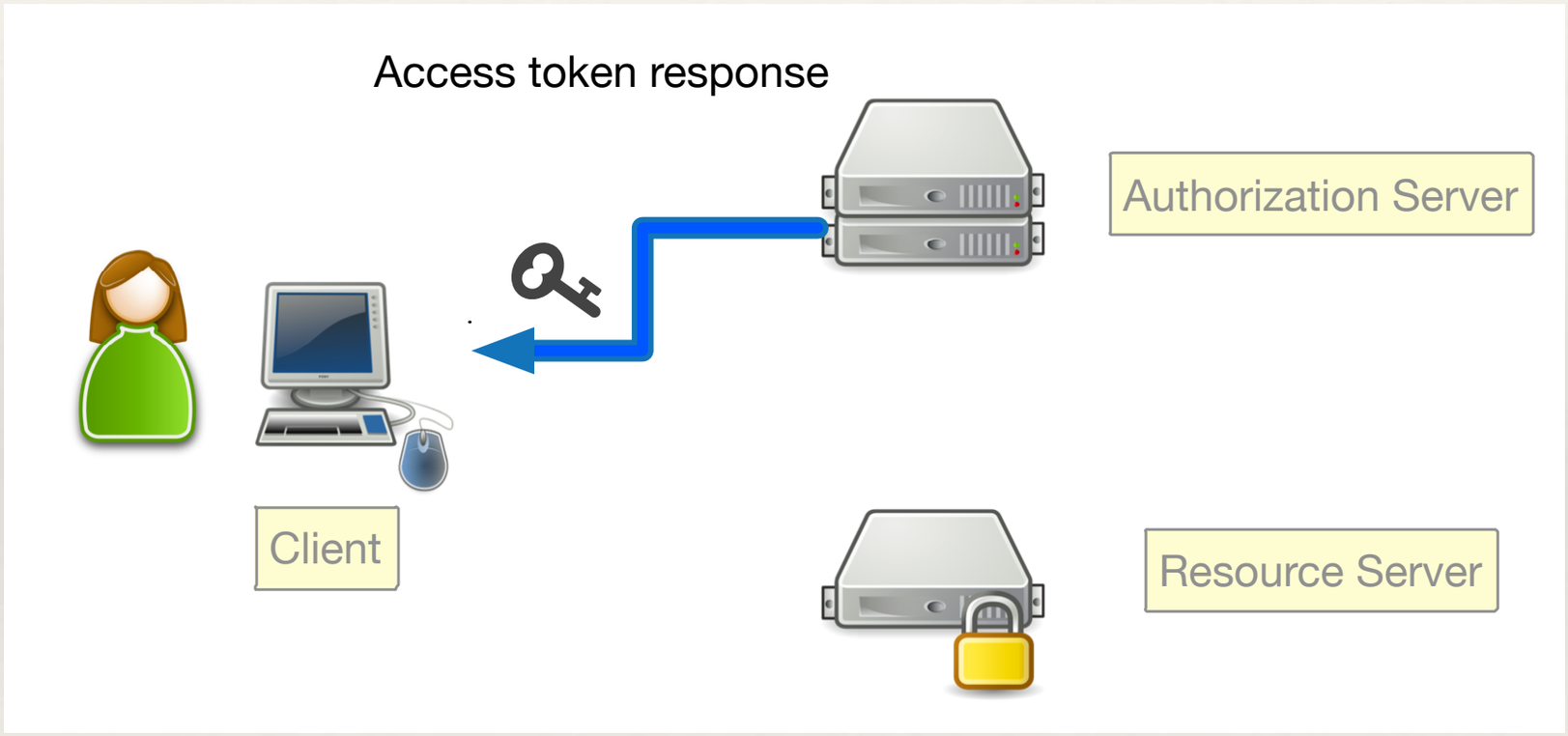
grant_type

redirect_uri




```
POST /token HTTP/1.1
Host: as.example.com
Content-Type: application/x-www-form-urlencoded
Authorization: Basic QWxhZGRpbjpPcGVuU2VzYW1l

code=s87BT60pp2UbNX2HnkWpZ9YhPVHRZaoTuU9XJul6JMuQaKUidUM6y1Boab6&
      grant_type=authorization_code&redirect_uri=https%3A%2F%2Fexample.org%2Fauthz_cb
```

Access Token Response



Access Token Response - details

Parameters

access_token

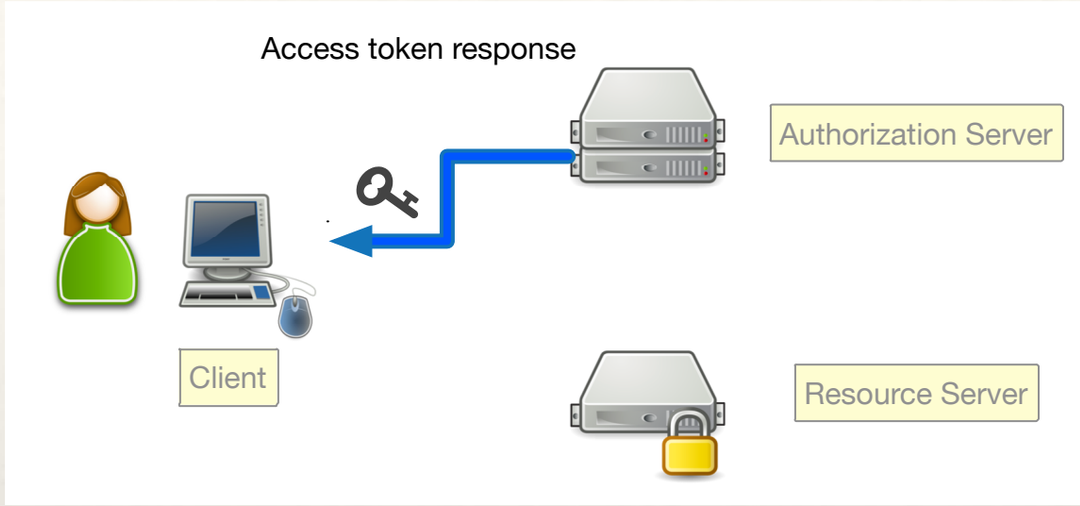
expires_in

refresh_token

scope

token_type

state




```
HTTP/1.1 200 OK
Content-Type: application/json;charset=UTF-8

{
        'access_token': 'bNX2HnkWpfPfWNo9Gi7chACuWoa2IDND',
        'expires_in': 3600,
        'refresh_token': 'bNX2HnkWpVCnDYPsy8EOpI'
        'state': 'STATE0',
        'token_type': 'Bearer',
}
```

Access Token

A string representing an authorization issued to the client. The string is usually opaque to the client.

Bearer Token

’A security token with the property that any party in possession of the token (a "bearer") can use the token in any way that any other party in possession of it can.’

Refresh Token

Refresh tokens are issued to the client by the authorization server and are used to obtain a new access token when the current access token becomes invalid or expires, or to obtain additional access tokens with identical or narrower scope.

Refresh Access Token Request - details

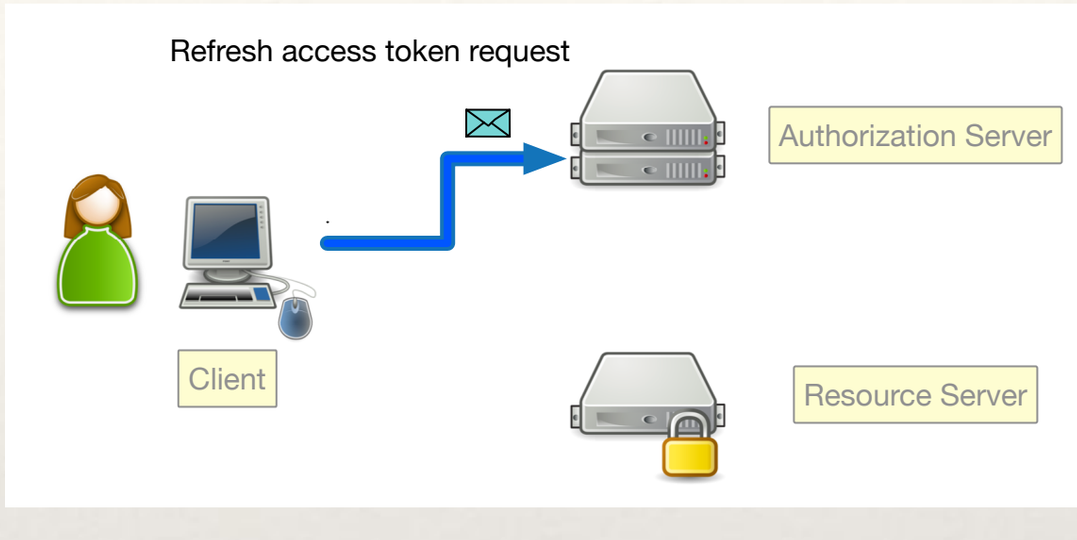
Parameters

client_id

grant_type

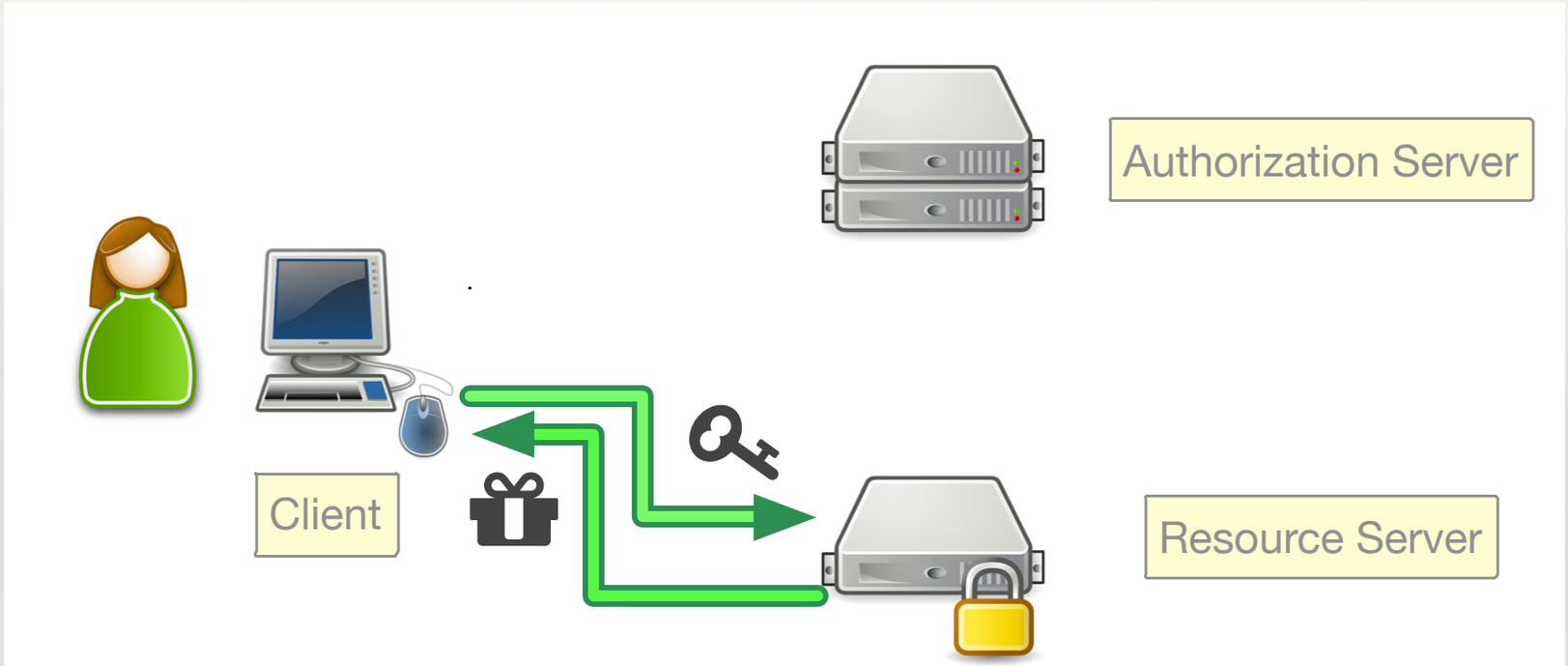
scope

refresh_token



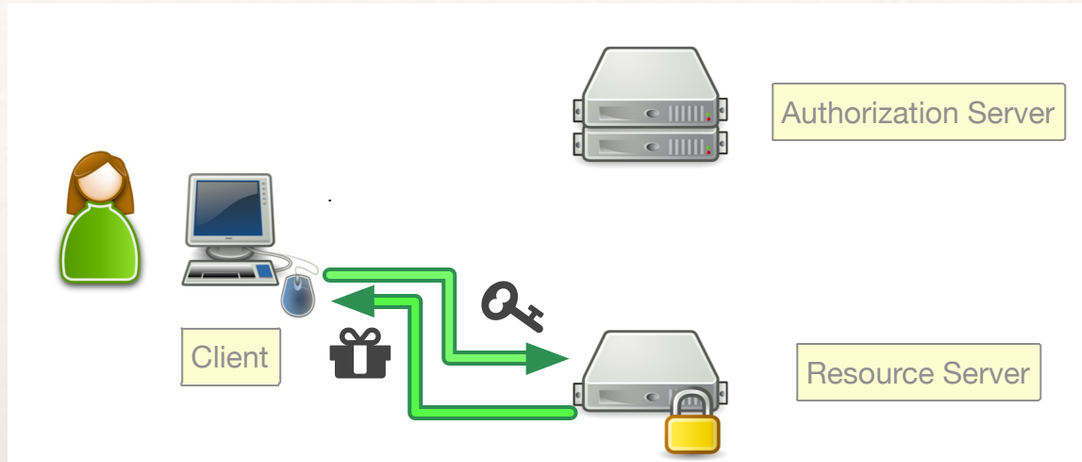

```
POST /token HTTP/1.1
Host: as.example.com
Authorization: Basic QWxhZGRpbjpPcGVuU2VzYW1l

refresh_token=bNX2HnkWpVCnDYPsy8EOpI&grant_type=refresh_token
```

Resource Access



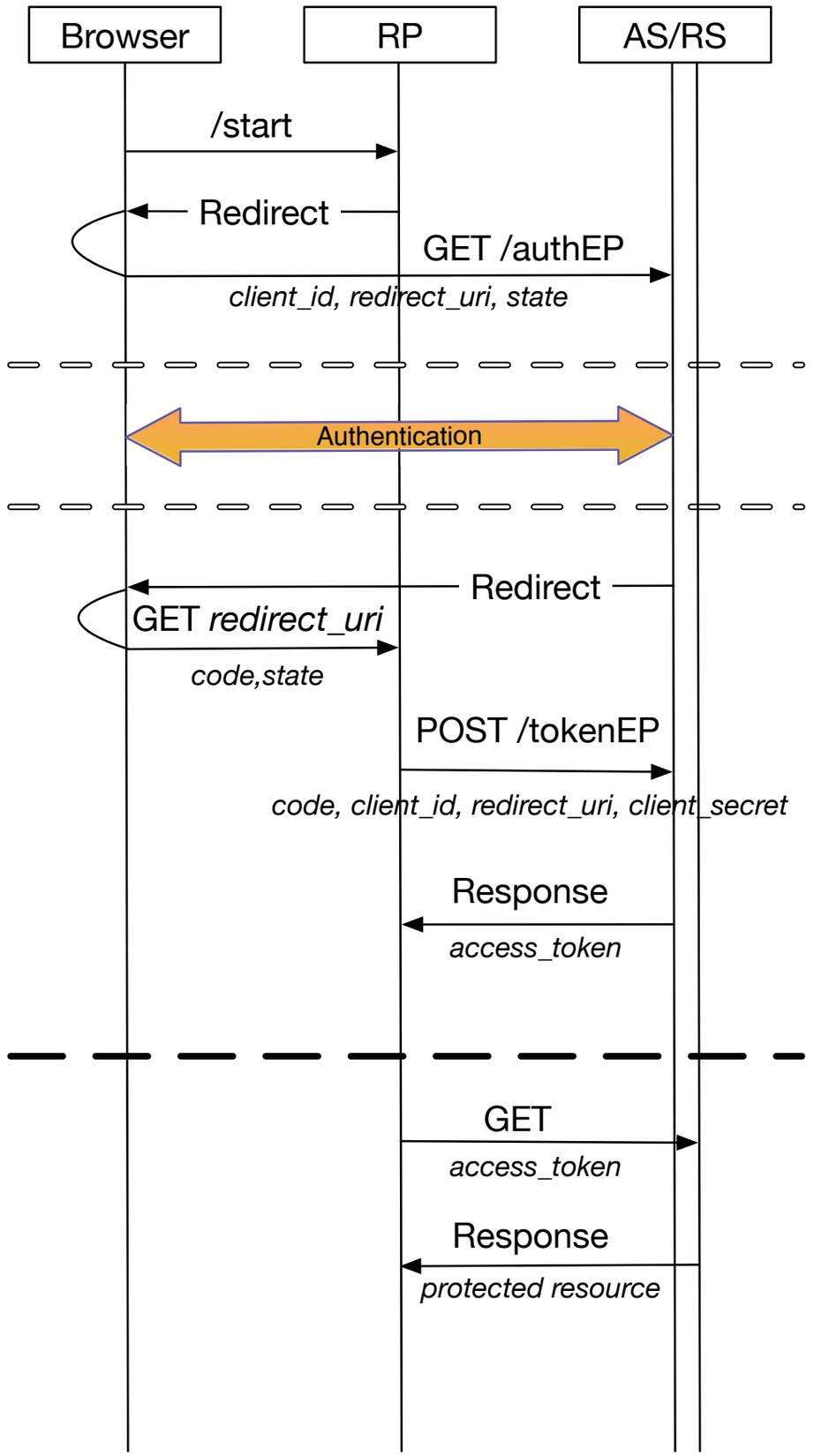
Resource request/response

Resource Access - details

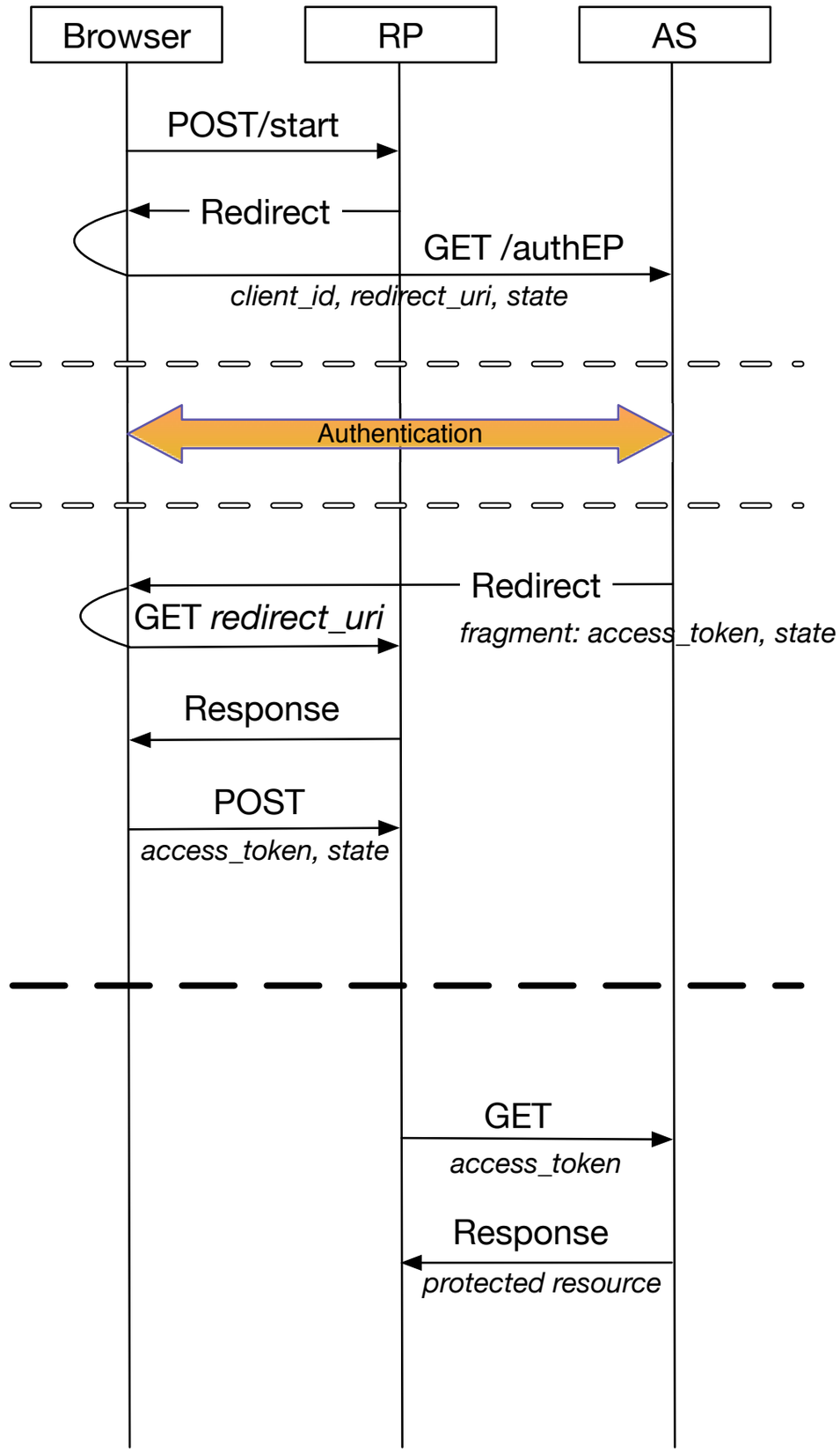



```
GET /resource HTTP/1.1
Host: rs.example.com
Authorization: Bearer bNX2HnkWpfPfWNo9Gi7chACuWoa2IDND
```

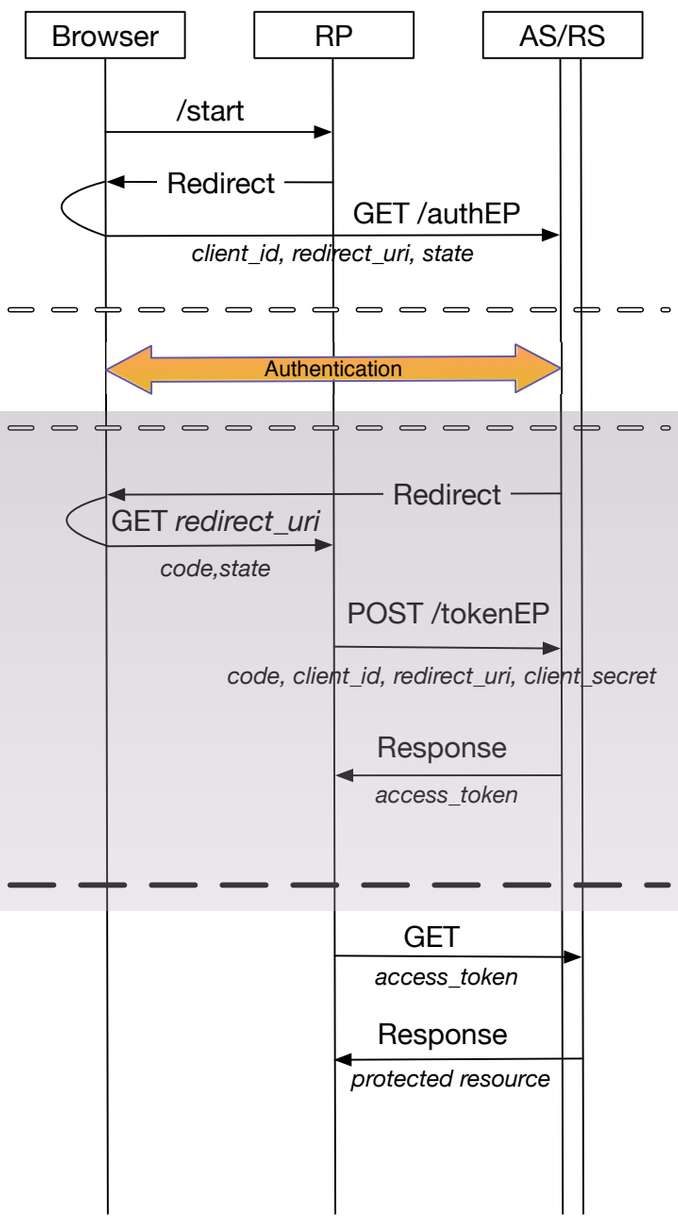
Code

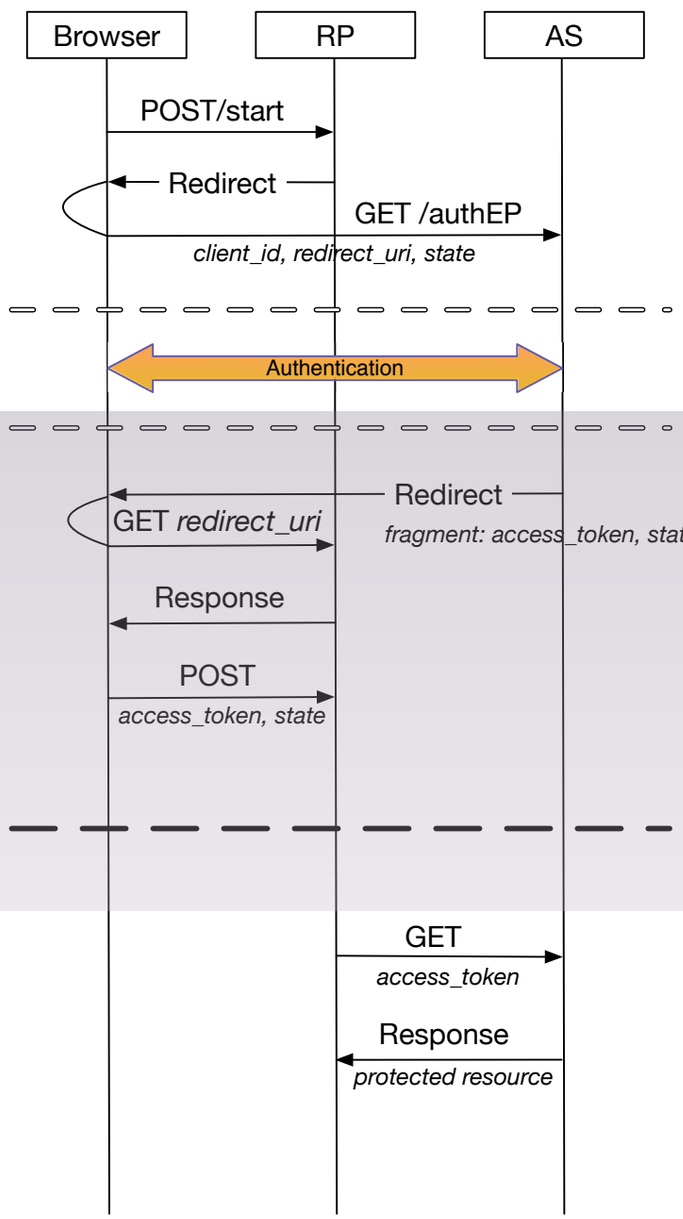

Browser
RP
AS/RS
/start
Redirect
GET /authEP
client_id, redirect_uri, state
Authentication
Redirect
GET redirect_uri
code,state
POST /tokenEP
code, client_id, redirect_uri, client_secret
Response
access_token
GET
access_token
Response
protected resource

Token

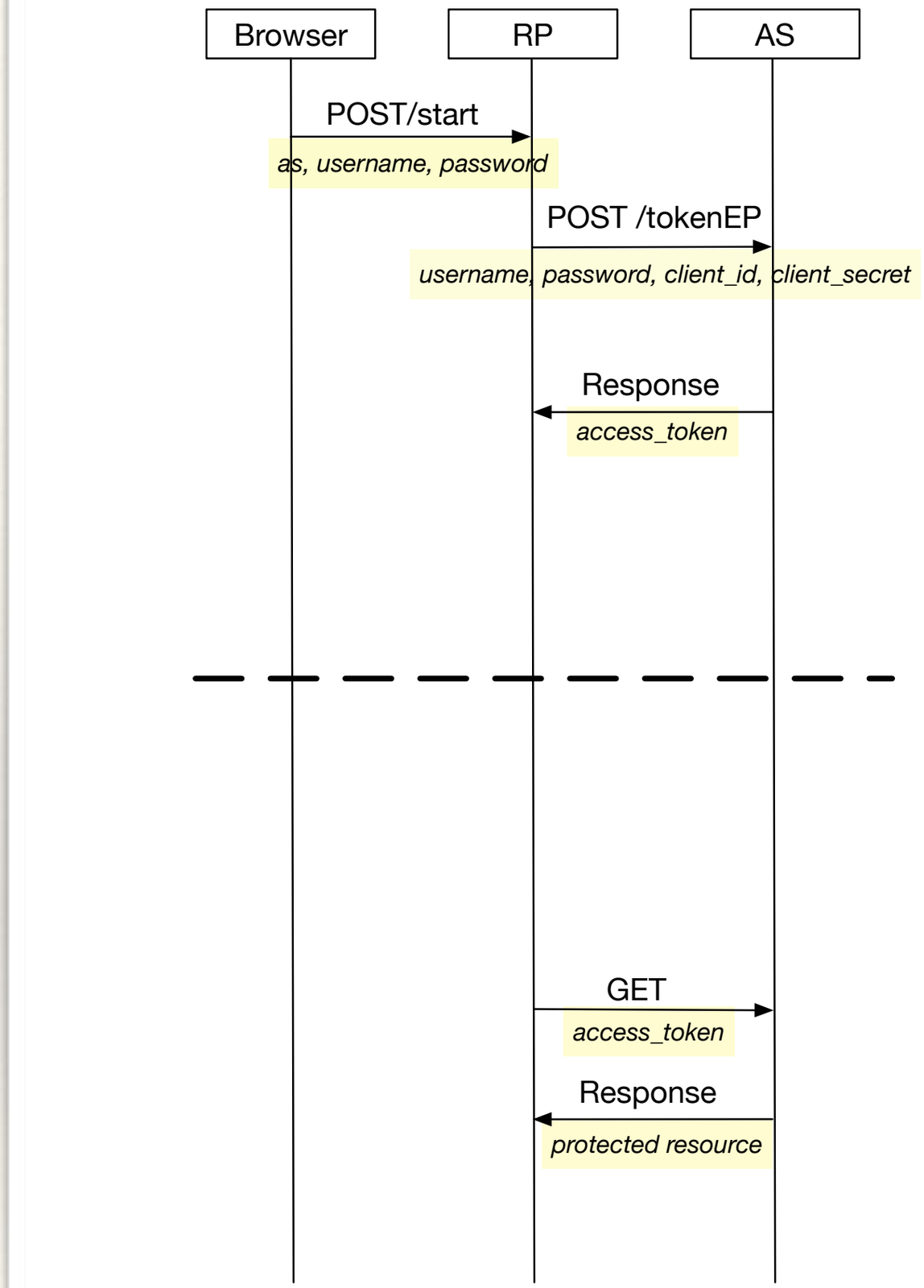

Browser
RP
AS
POST/start
Redirect
GET /authEP
client_id, redirect_uri, state
Authentication
Redirect
GET redirect_uri
fragment: access_token, state
Response
POST
access_token, state
GET
access_token
Response
protected resource

Code vs Implicit

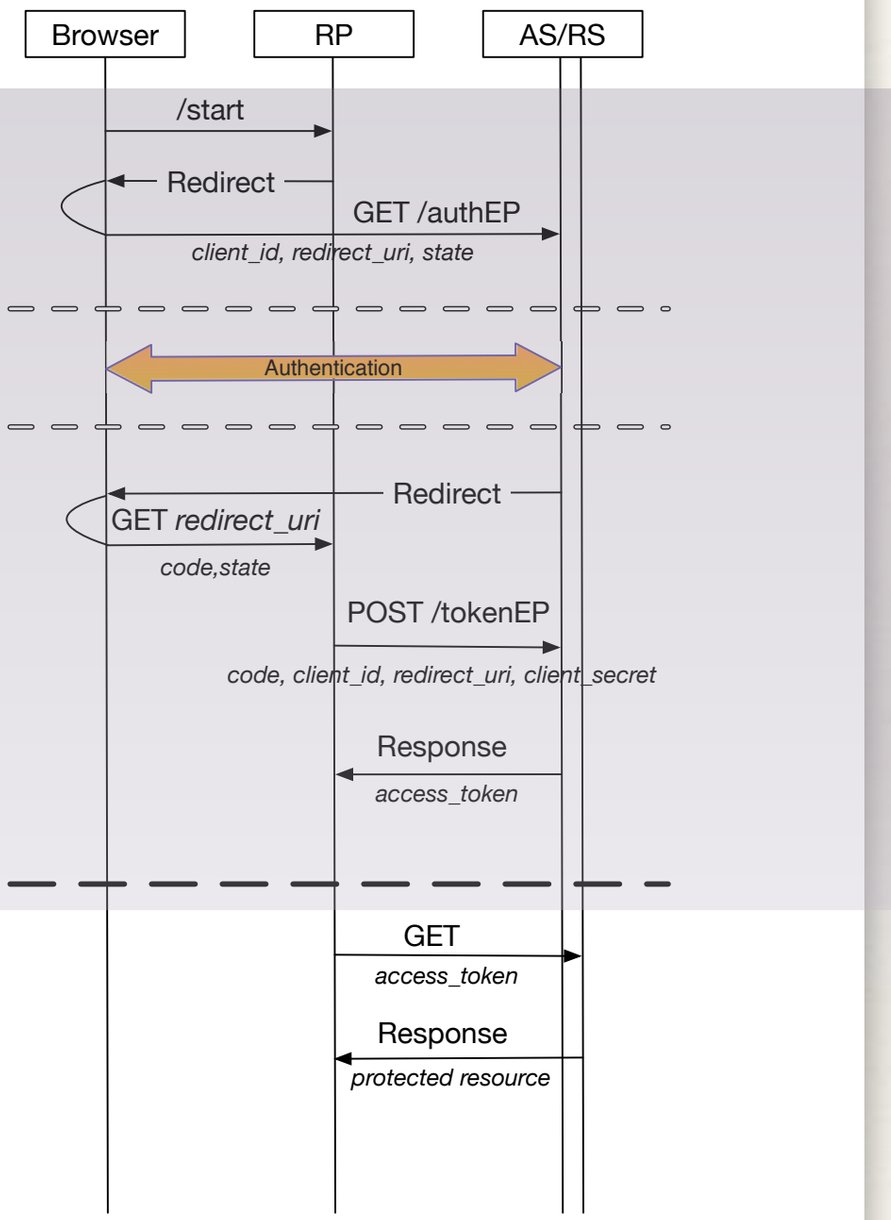


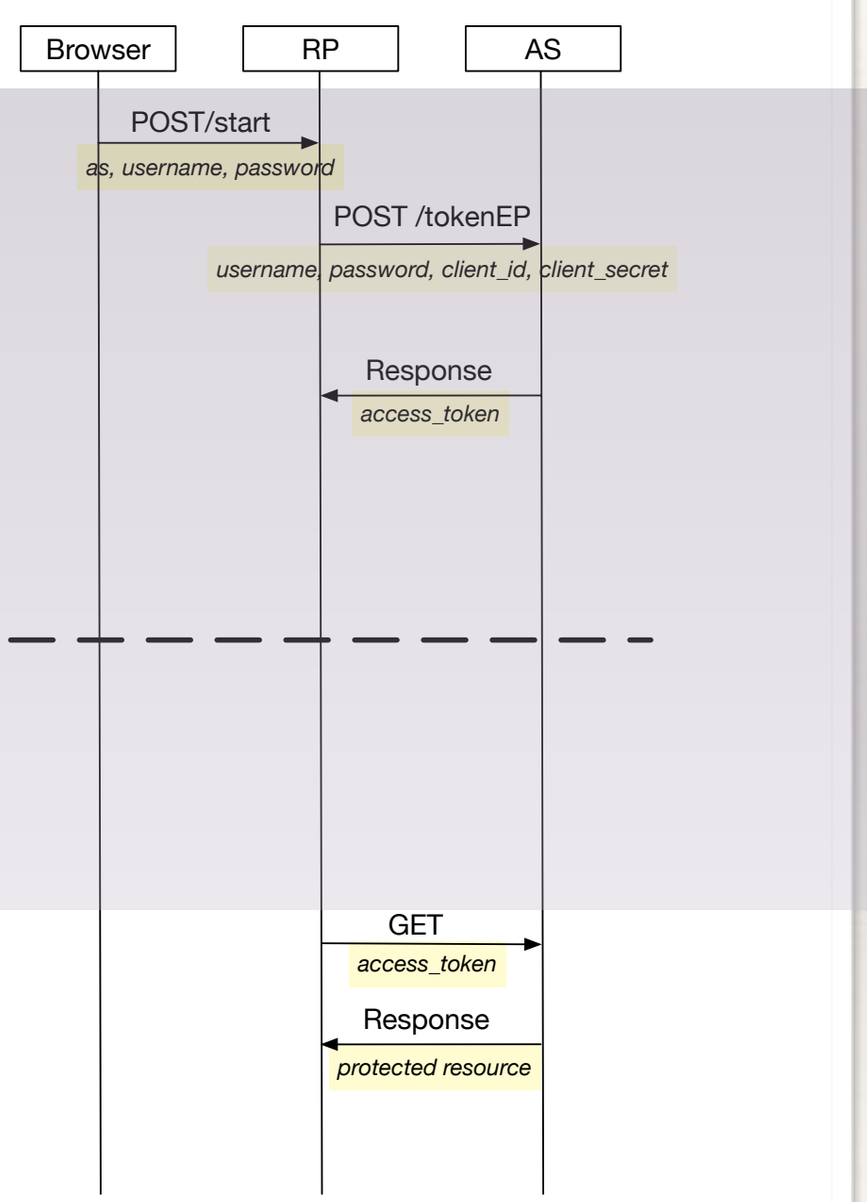


Resource Owner Password Credentials

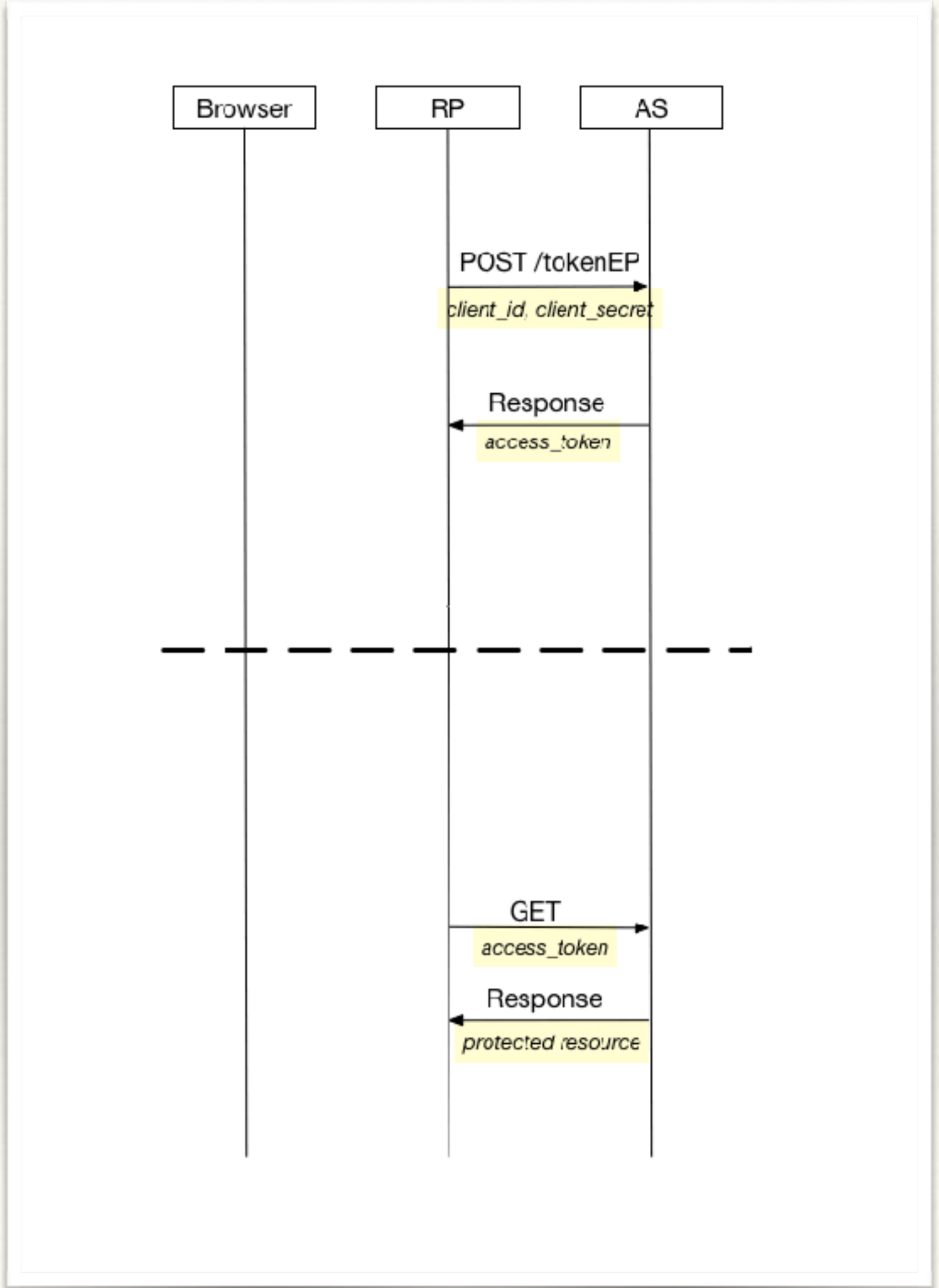


Code vs ROPC

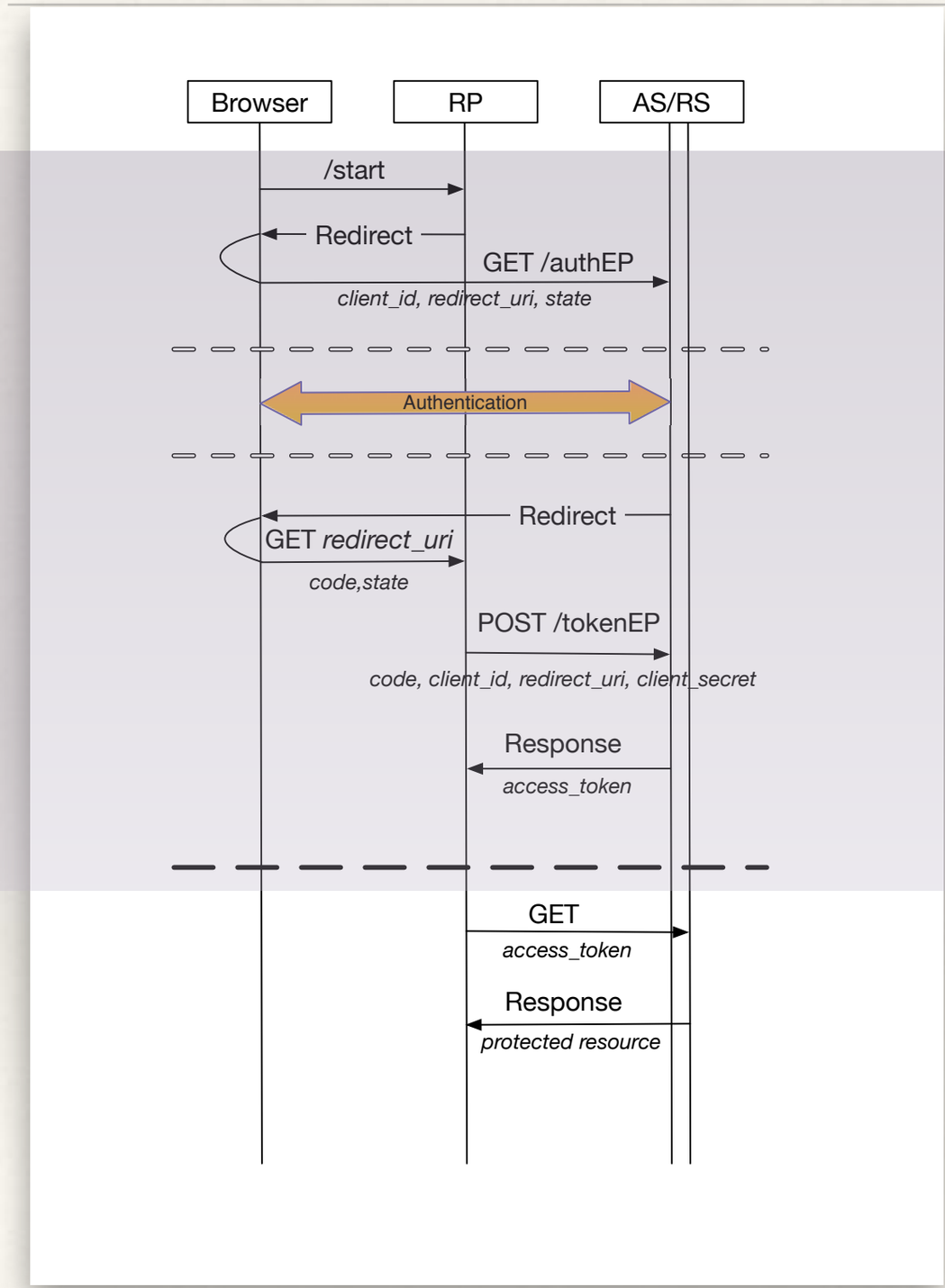

Browser
RP
AS/RS
/start
Redirect
GET /authEP
client_id, redirect_uri, state
Authentication
Redirect
GET redirect_uri
code,state
POST /tokenEP
code, client_id, redirect_uri, client_secret
Response
access_token
GET
access_token
Response
protected resource

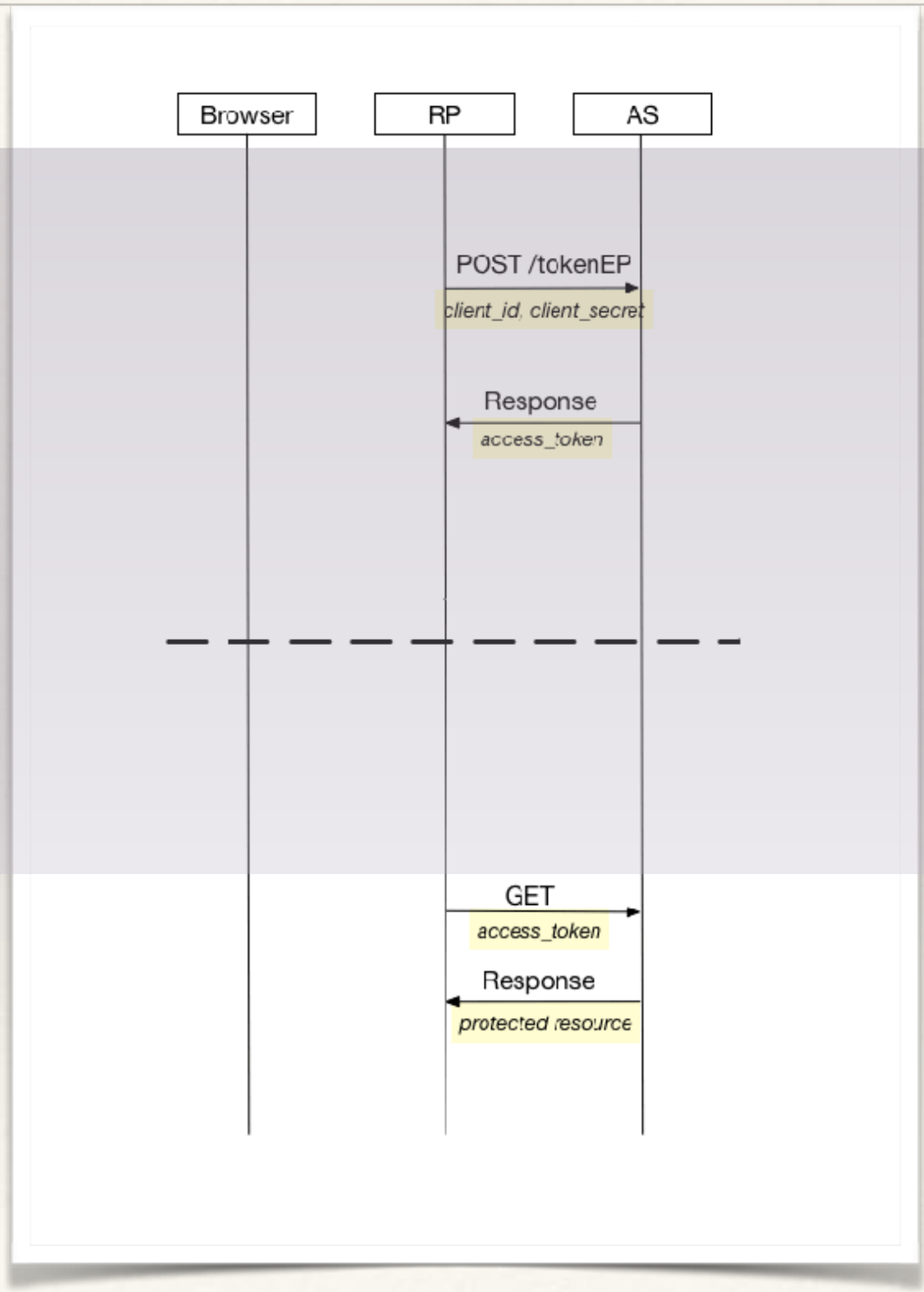

Browser
RP
AS
POST/start
as, username, password
POST /tokenEP
username, password, client_id, client_secret
Response
access_token
GET
access_token
Response
protected resource

Client Credentials



Code vs CliCred





Note

Information is transmitted URL-encoded or as a JSON document

Links to documents

- The OAuth 2.0 Authorization Framework (RFC6749)
- The OAuth 2.0 Authorization Framework: Bearer Token Usage (RFC6750)