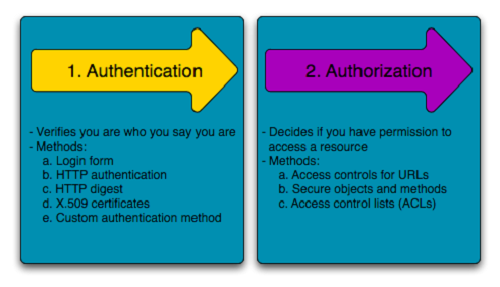
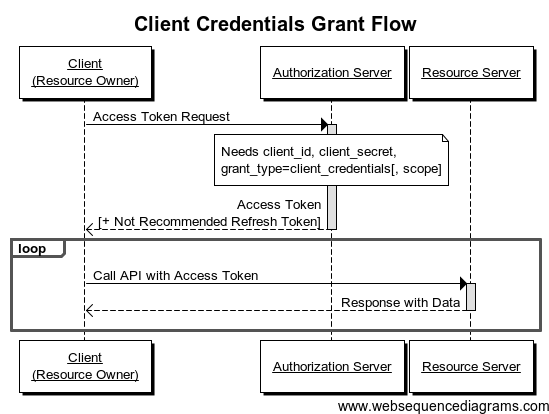
**API Gateway Design Pattern**

**Implemented:**

* Authentication
* Authorization
* API Gateway (Netflix ZUUL API gateway)
* Eureka Naming server (service discovery)
* Authorization server (oauth2 demo, generates JWT token after successful authentication, authorization)
* 3 microservices (2 need authorization, 1 service can be called without authorization and authentication (like home page services, etc.))
* Services as ademp-area-service, claims-area-service, prints-area-service (doesn’t require authorization and authentication through API Gateway)







**CASE 1: Implementation - done**

Authorization server  
here it will check if the user if user is present In memory (authentication)

After it will check the authorization using passed client id and client secret (authorization)  
  
After successful authentication and authorization it will return token (JWT)

PRINTS area Service

CLAIMS area Service

ADEMP area Service

Authorization with token

API Gateway also acts as resource server in our case

**CASE 2: Implementation – done**

Authorization server  
here it will check if the user if user is present In memory (authentication)

After it will check the authorization using passed client id and client secret (authorization)  
  
After successful authentication and authorization it will return token (JWT)

Authorization with token

PRINTS area Service

CLAIMS area Service

ADEMP area Service

API Gateway also acts as resource server in our case