Securing Microservices

Chapter Content

- OAuth2 Intro

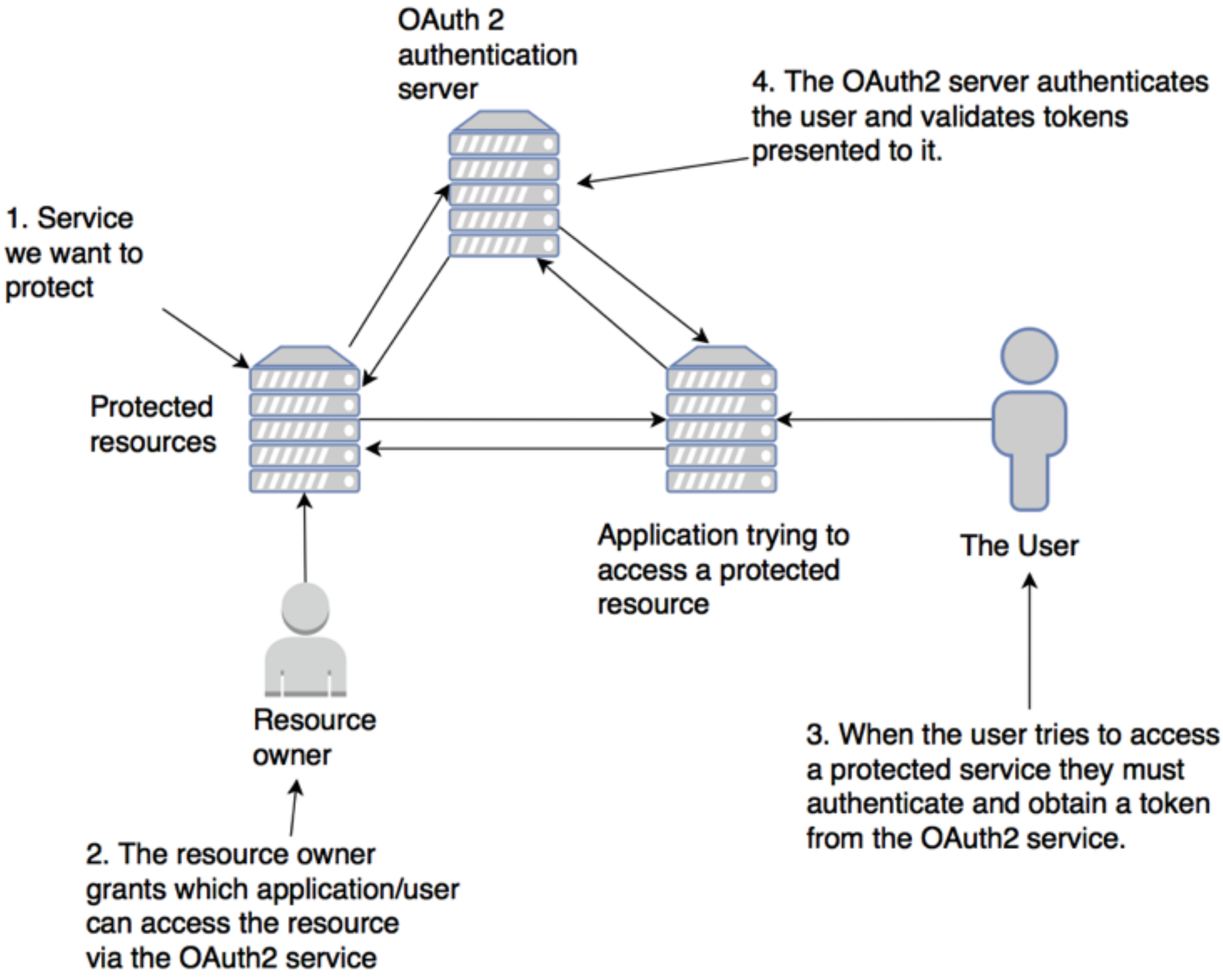
- A secure application involves multiple layers of protection
    - User Authentication and Authorization
    - Keep the infrastructure patched and up-to-date
    - Network access control so that a service is only accessible through well-defined ports and accessible to a small number of authorized servers

- We will use Spring Cloud Security and OAuth 2

- OAuth2 is a token-based security framework that allows a user to authenticate themselves with a third-party authentication service

- If the user successfully authenticates, they will be presented a token that must be sent with every request. The token can then be validated back to the authentication service.

- The main goal behind OAuth2 is that when multiple services are called to fulfill a user's request, the user can be authenticated by each service without having to present their credentials to each service processing their request


OAuth 2
authentication
server
4. The OAuth2 server authenticates
the user and validates tokens
presented to it.
1. Service
we want to
protect
Protected
resources
Application trying to
access a protected
resource
The User
Resource
owner
3. When the user tries to access
a protected service they must
authenticate and obtain a token
from the OAuth2 service.
2. The resource owner
grants which application/user
can access the resource
via the OAuth2 service

- OAuth2 allows you to protect your REST-based services across these different scenarios through different authentication schemes:
    - Password
    - Client credential
    - Authorization Code
    - Implicit