Лекция 26

ПСКП, ПОИТ-3

**Авторизация (Authorization)**

1. **Идентификация** – заявление пользователя о себе.
2. **Аутентификация** – процедура проверки подлинности идентификации пользователя.
3. **Авторизация** -процедура проверки прав аутентифицированного пользователя.
4. **Authorization:** процедура предоставления доступа к определенному ресурсу/действию на основании каких-то признаков (флагов, ролей, привилегий и т.д.).
5. **Authorization:** виды:

а) дискреционное управление доступом (DAC);

б) мандатное управление доступом (MAC);

в) управление доступом на основе ролей (RBAC).

1. **Authorization:** дискреционное (прямое) управление доступом (DAC): список доступа (ACL) субъектов к объекту (Microsoft Windows, Unix)

***Например:***

пользователю user\_1 разрешено читать файл file\_1

пользователю user\_2 разрешено читать и писать в файл file\_1

1. **Authorization:** мандатное управление доступом (MAC): уровень доступа субъекта ≥ уровень секретности объекта

***Например:***

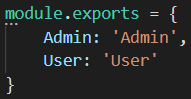
пользователю с уровнем доступа L1 разрешено читать файл file\_1

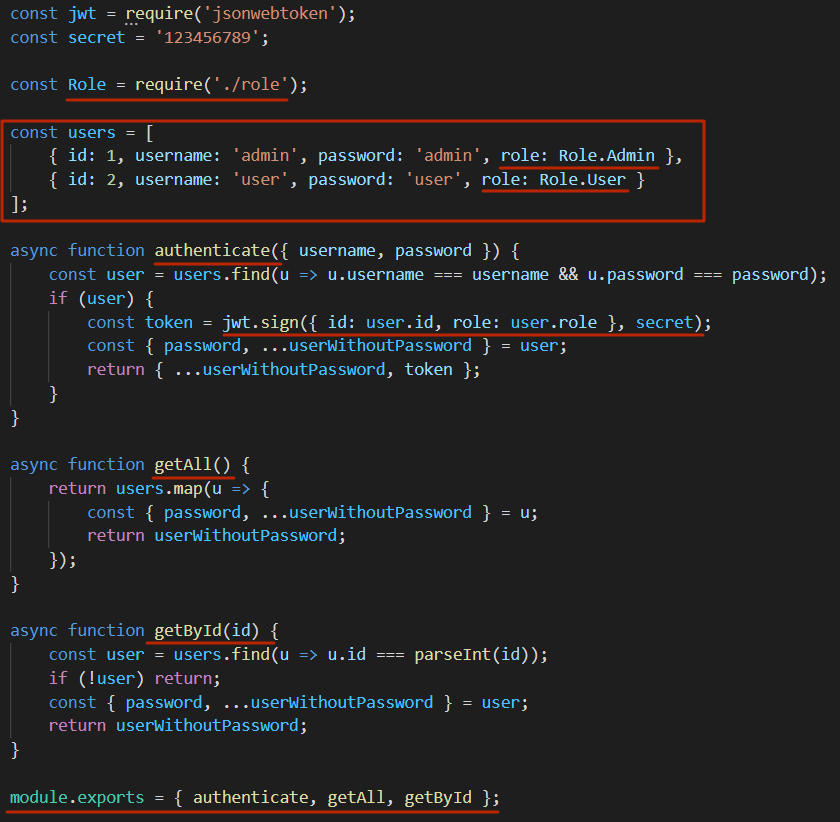
1. **Authorization:** управление доступом на основе ролей (RBAC): роль у субъекта, определяющая привилегии относительно объекта (Oracle, Microsoft Active Directory, PostreSQL)

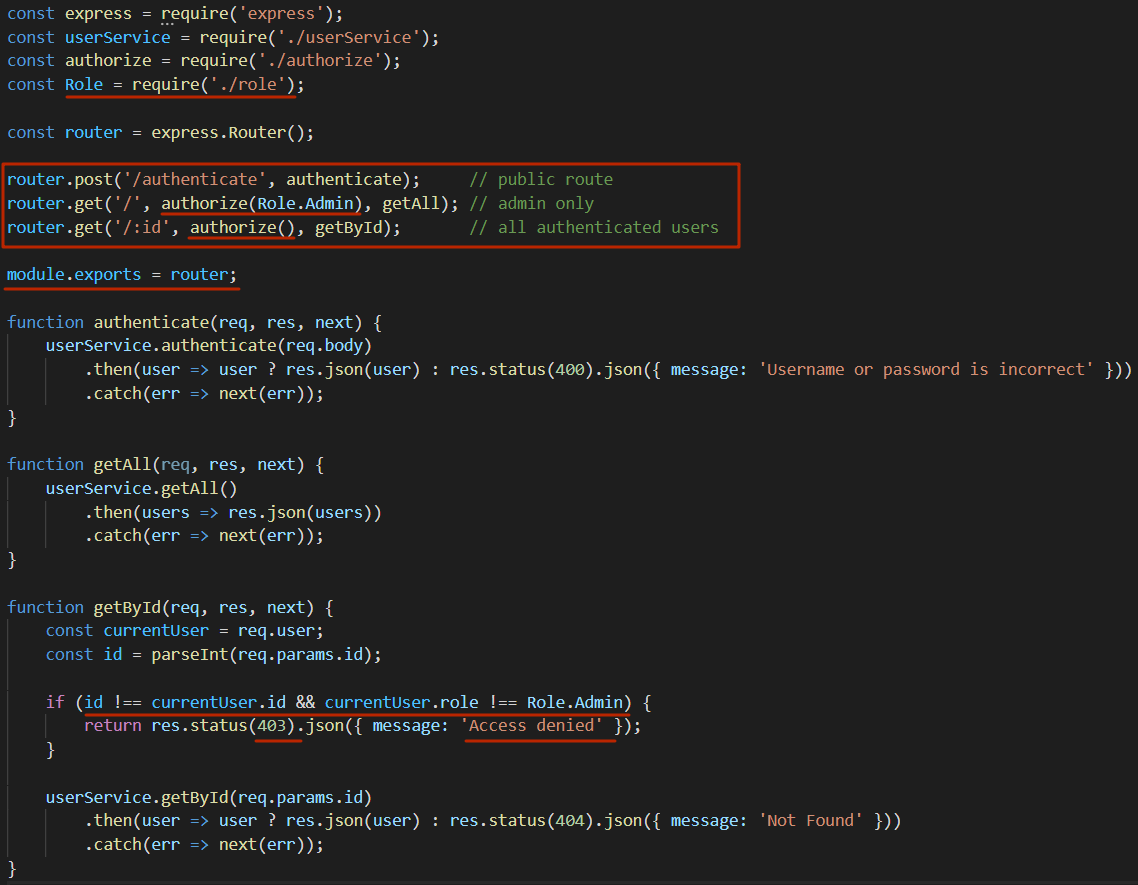
***Например:***

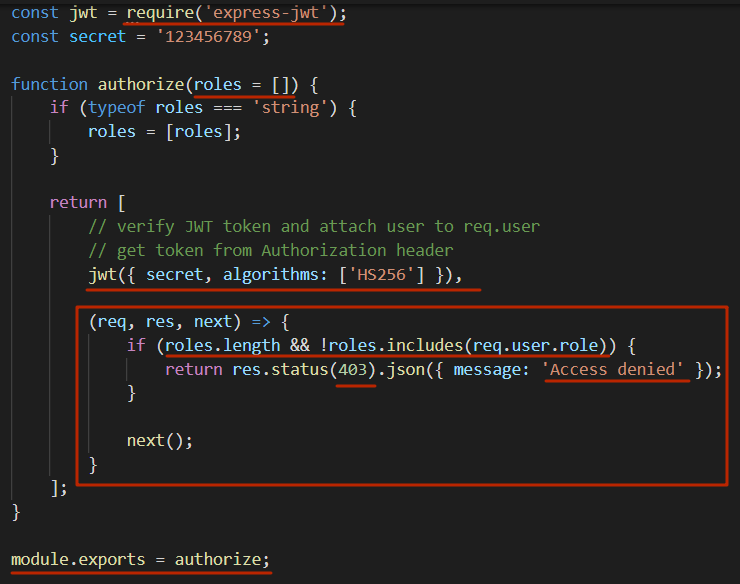
пользователю с ролью role\_1 разрешено читать файл file\_1

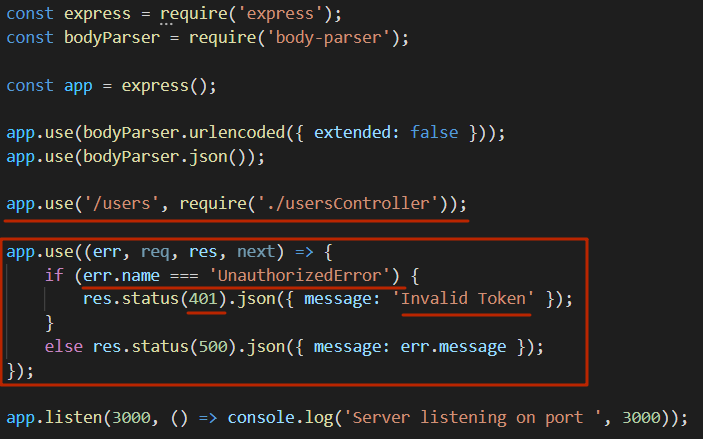
1. **Authorization:** роль - набор правил, определяющих, какими привилегиями и по отношению к каким объектам будет обладать пользователь, которому будет назначенаэта роль.
2. **Authorization:** привилегия (операция) — полномочие субъекта относительно какого-то объекта.
3. **Authorization:** самый распространенный способ **RBAC (+ACL)**
4. **Authorization:**

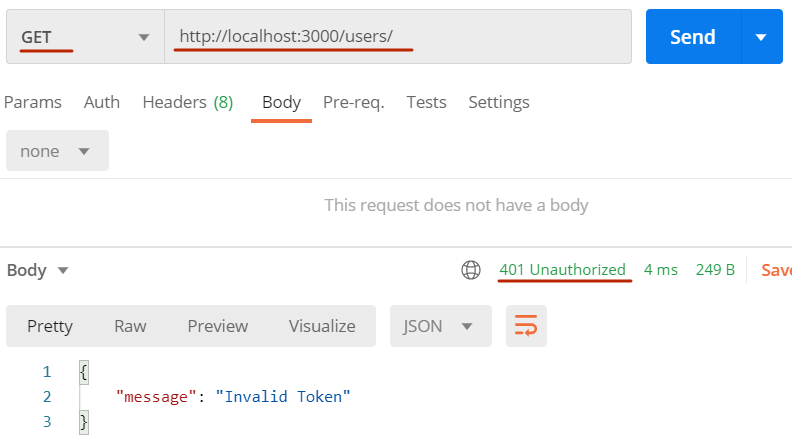


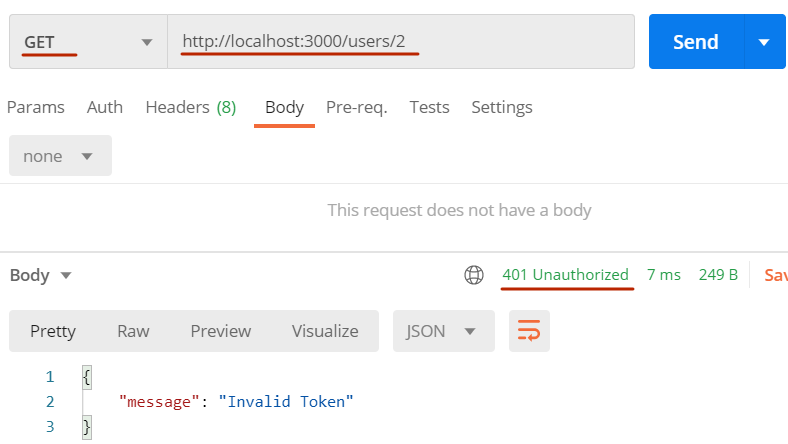


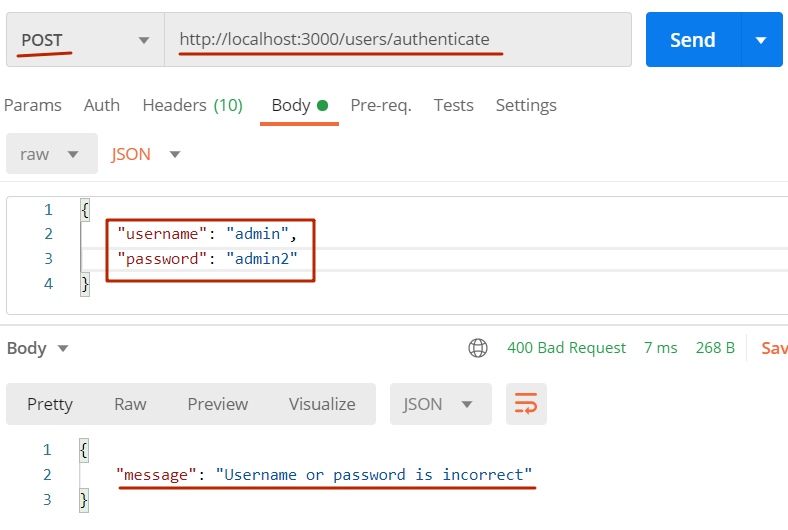


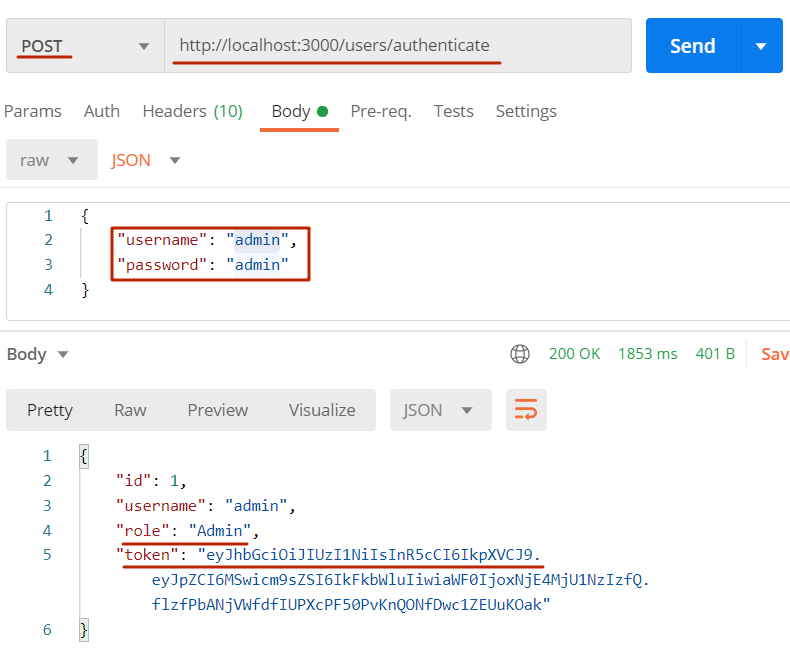


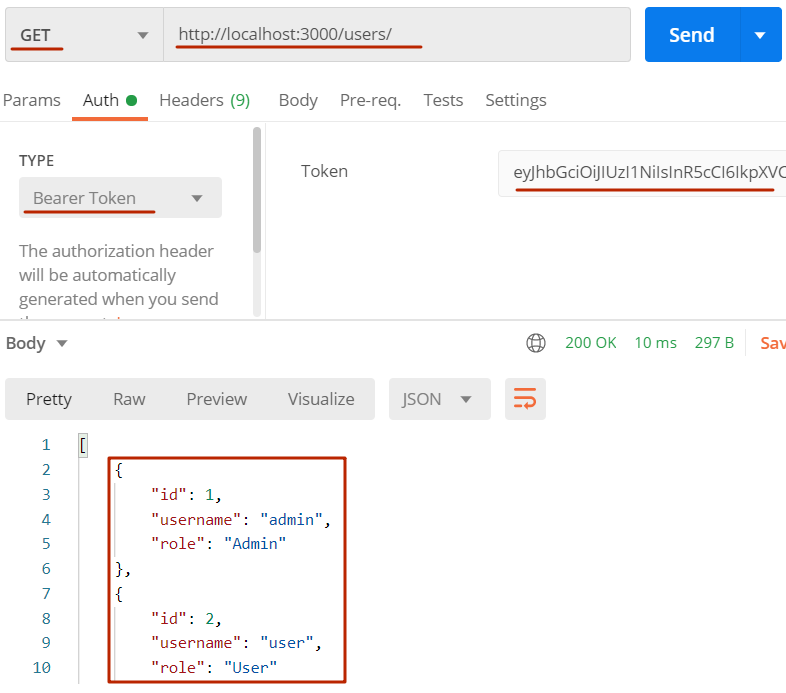


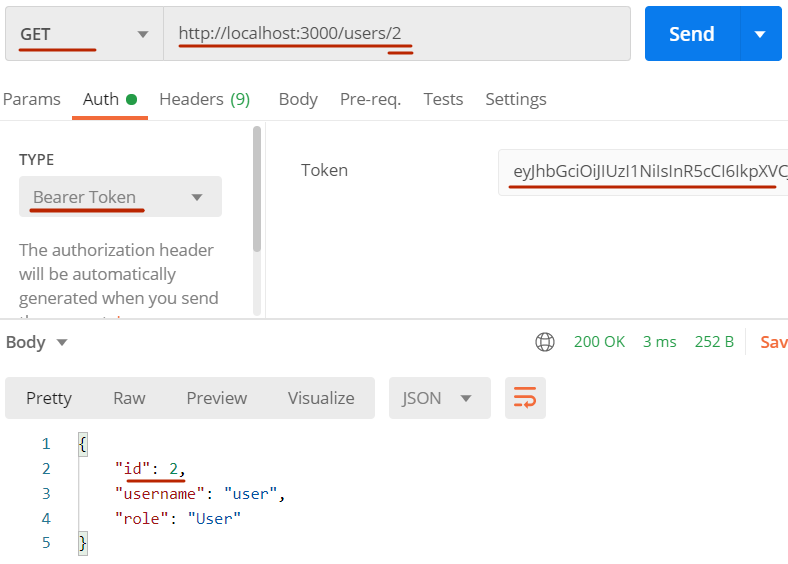


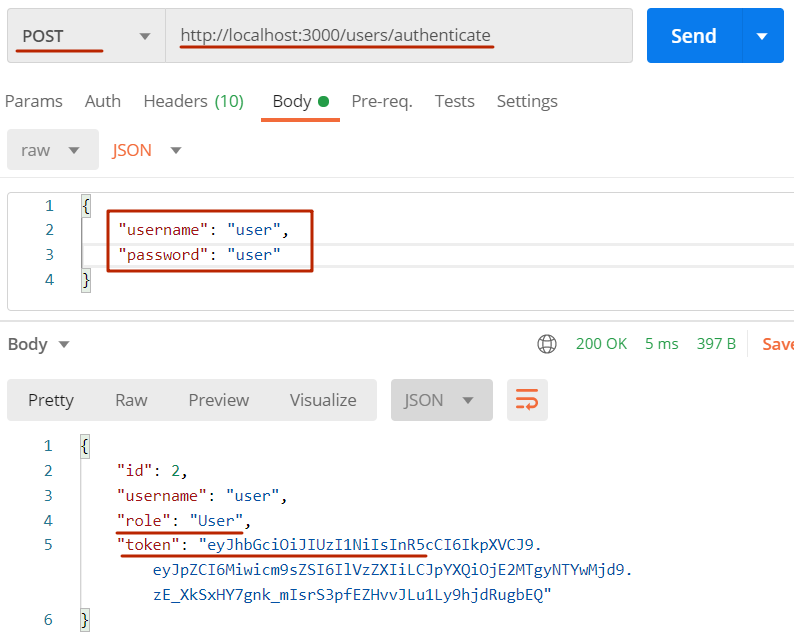


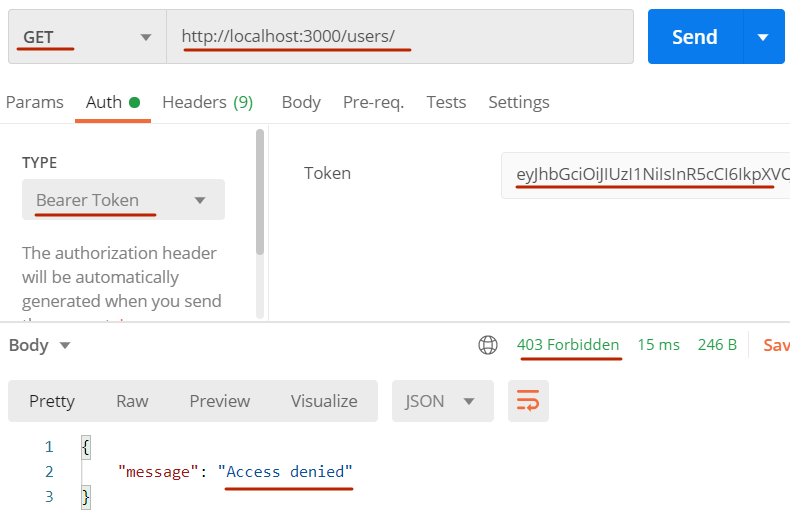


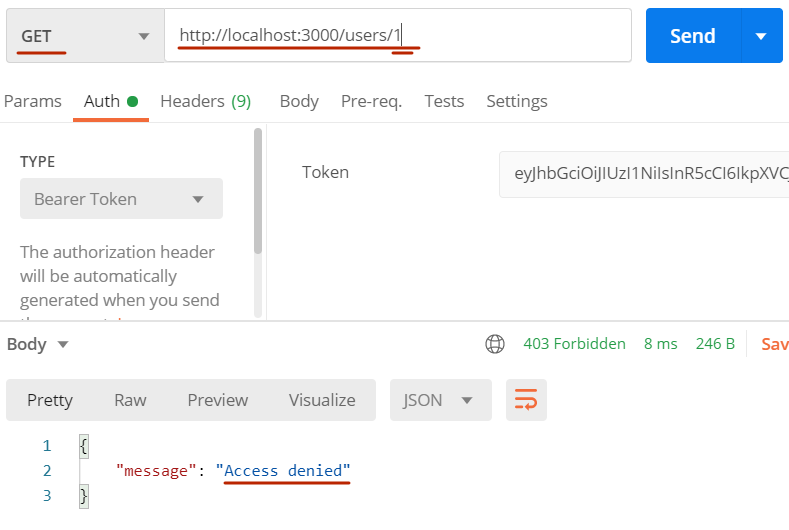


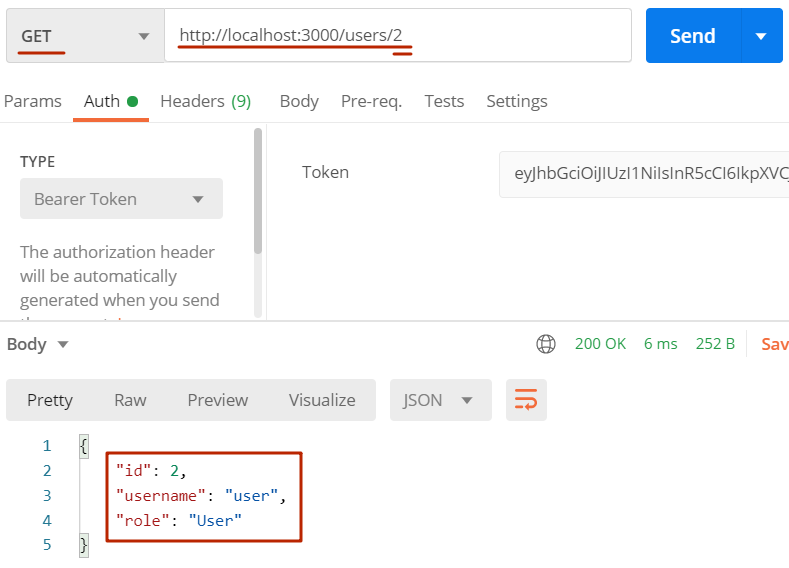


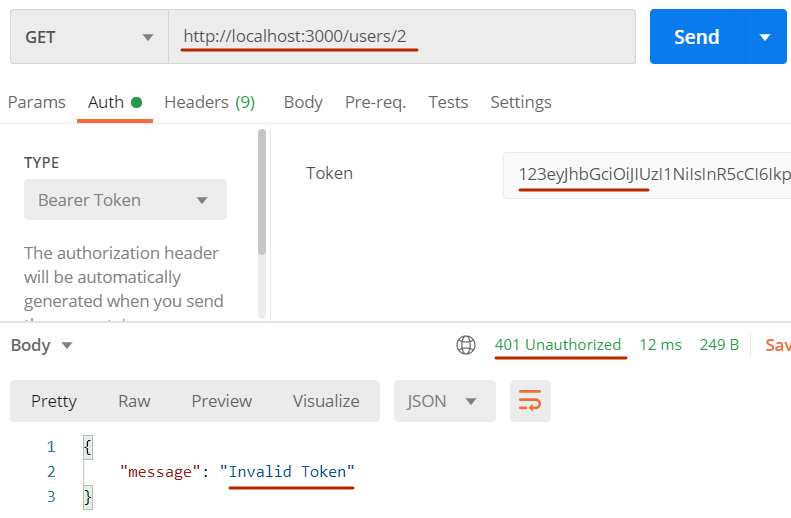




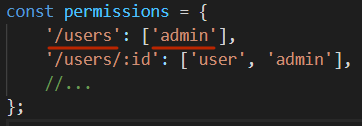


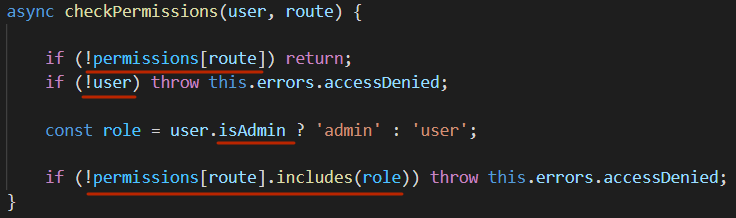


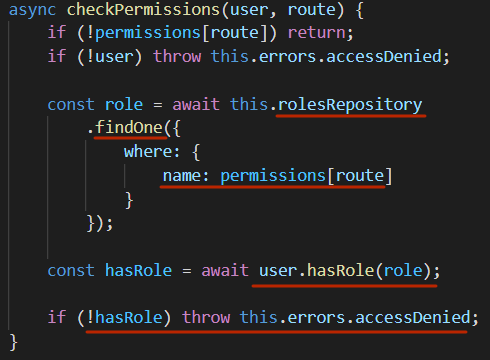




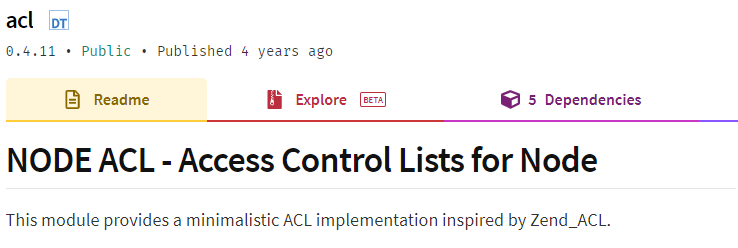
1. **Authorization:** ACL (Access Control Lists)







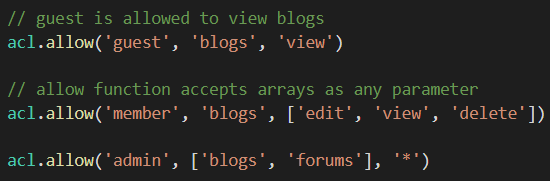
1. **NODE ACL:**



1. **NODE ACL:** npm install acl

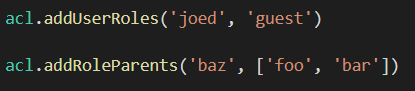


1. **NODE ACL:** allow(roles, resources, permissions, function(err))

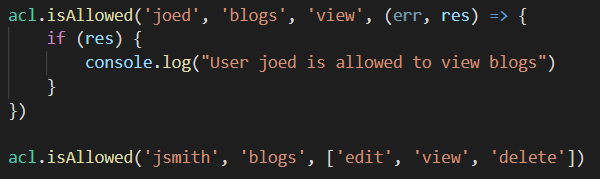


1. **NODE ACL:** addUserRoles(userId, roles, function(err))

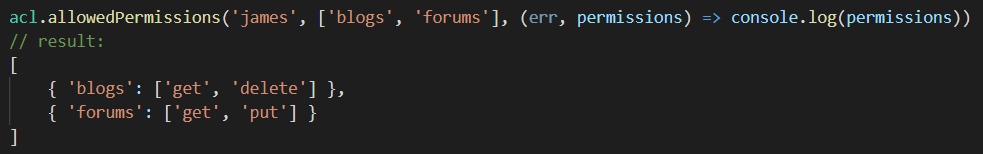
addRoleParents(role, parents, function(err))



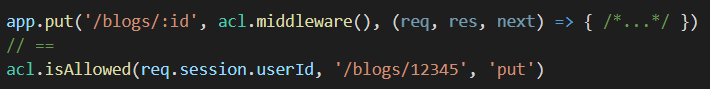
1. **NODE ACL:** isAllowed(userId, resource, permissions, function(err, allowed))



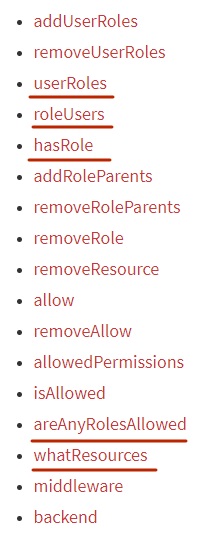
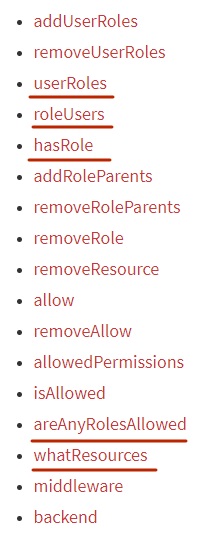
1. **NODE ACL:** allowedPermissions(userId, resources, function(err, obj))



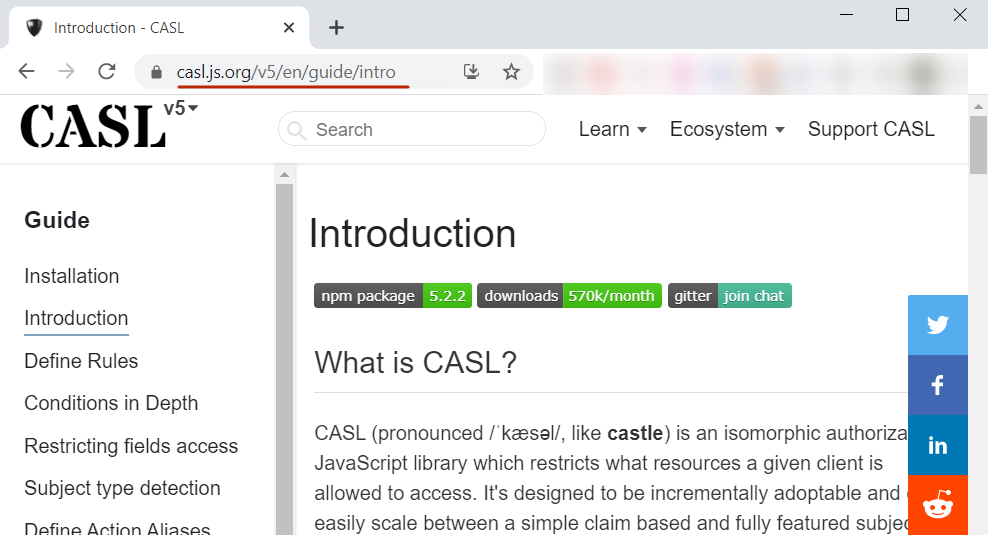
1. **NODE ACL:** middleware([numPathComponents, userId, permissions])



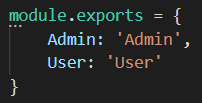
1. **NODE ACL:**

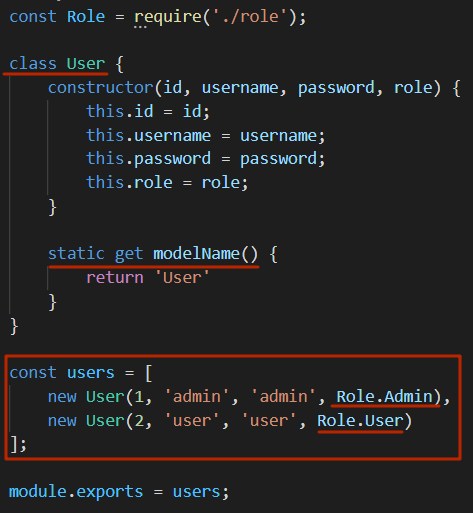
 

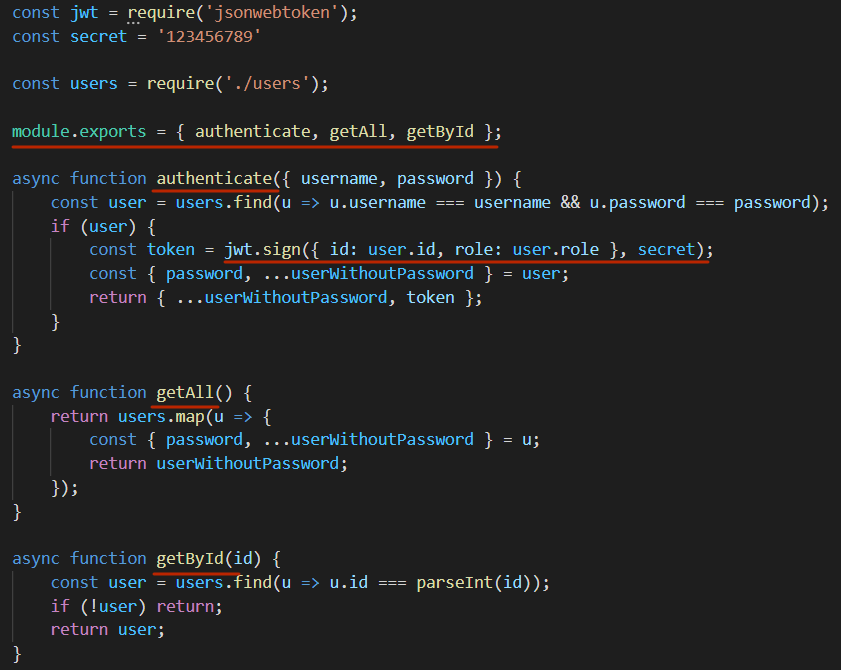
1. **CASL:** <https://casl.js.org/v5/en/guide/intro>

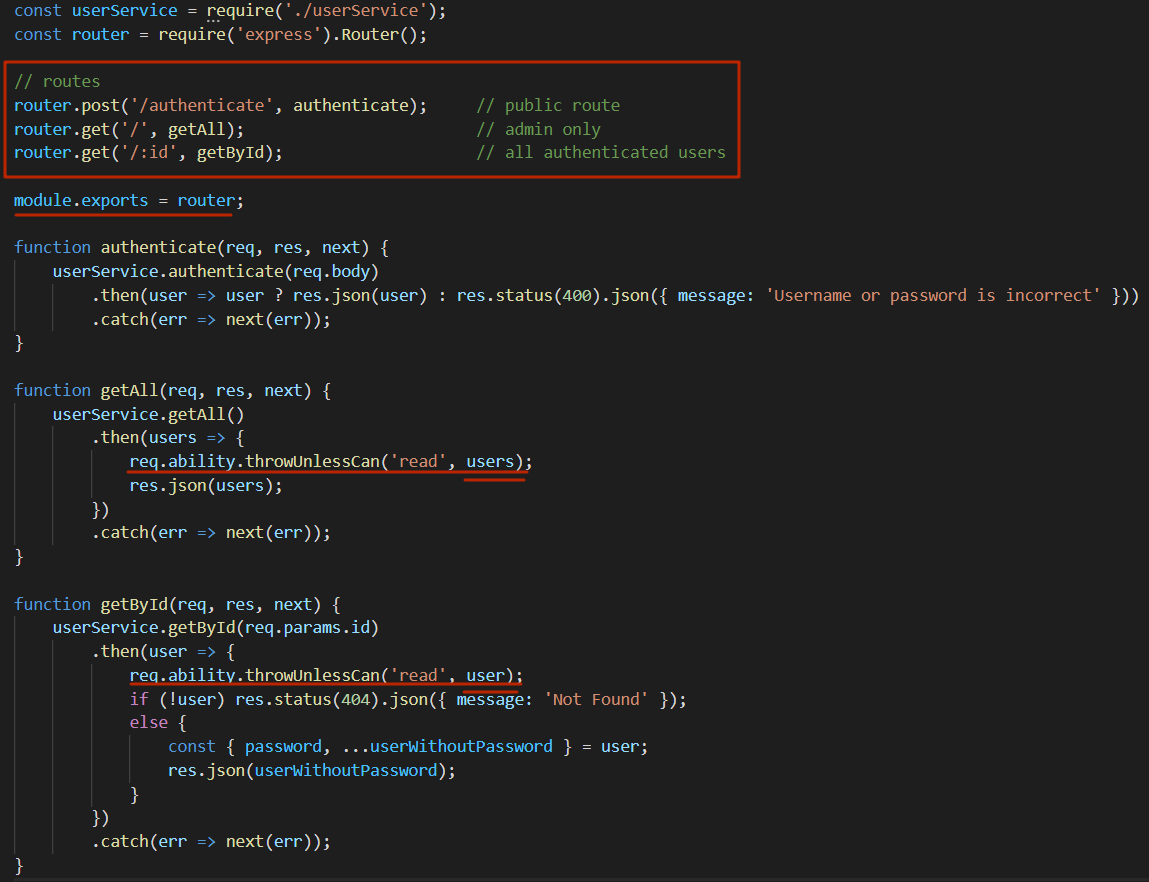


1. **CASL:**

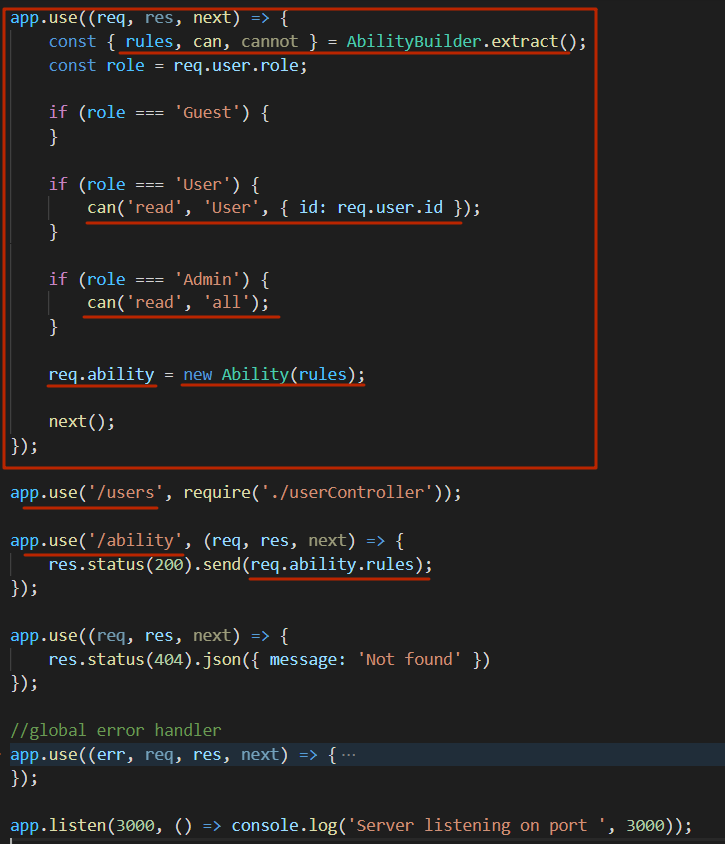


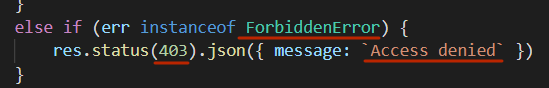


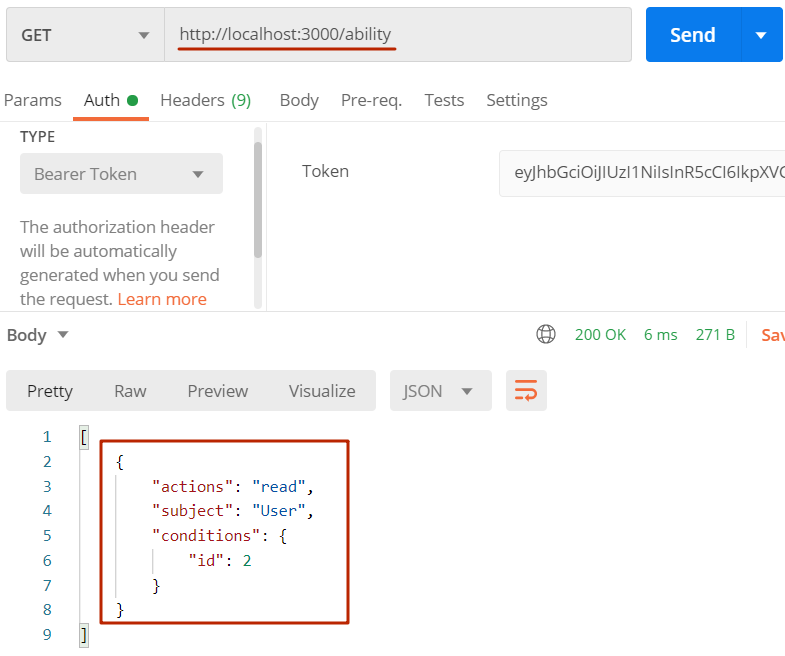


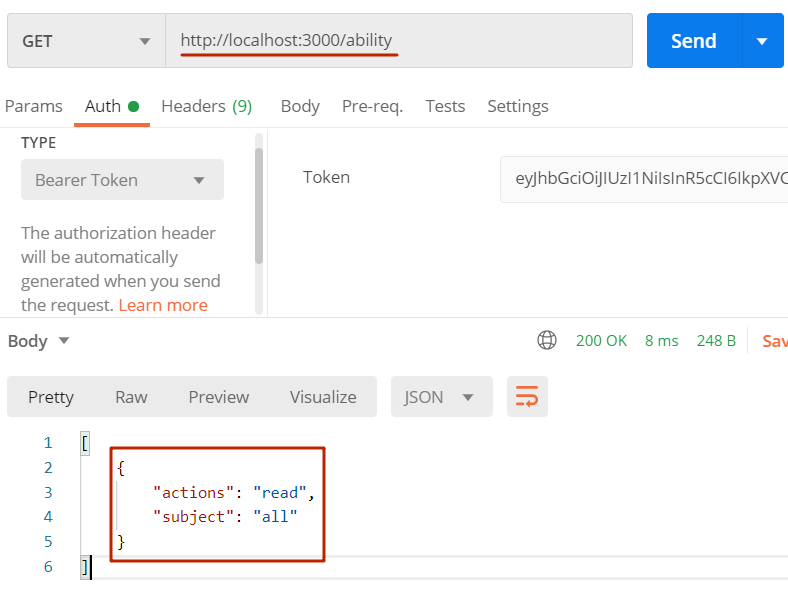


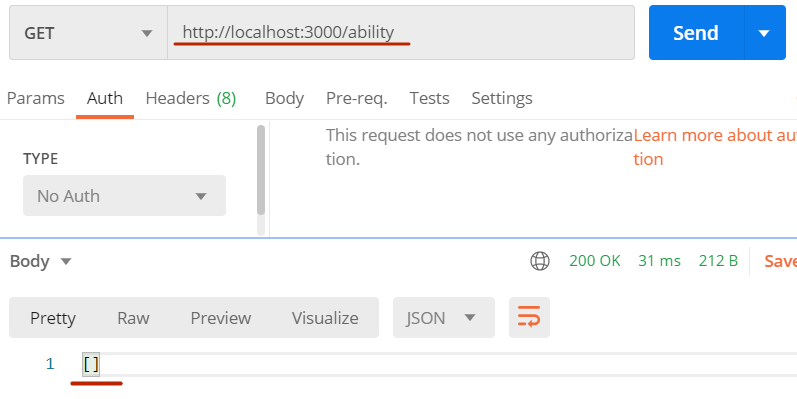




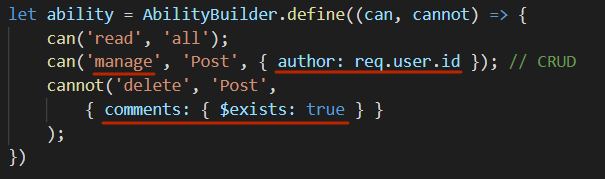


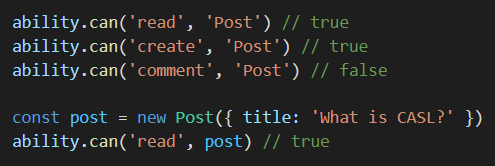






1. **CASL:**

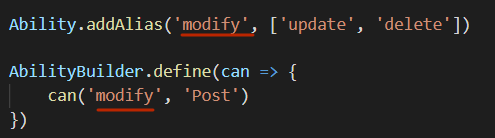




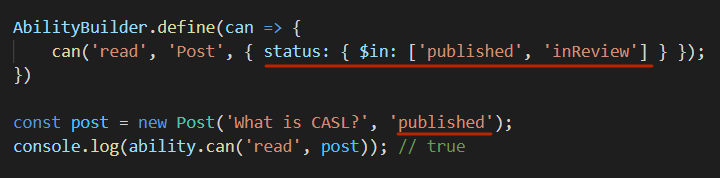
1. **CASL:**



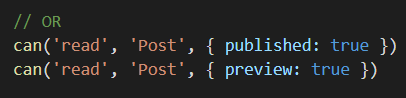
1. **CASL:**



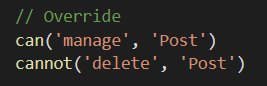
1. **CASL:** $exists, $in, $gte, $lte, $all, $regex…



1. **CASL:**



1. **CASL:**



1. **CASL:**

