48. Авторизация
часть 2

Цель:

Познакомиться со следующими особенностями авторизации:

- структура JWT

JWT:

JSON Web Token (**JWT**) — это открытый стандарт (RFC 7519) для создания токенов доступа, основанный на формате JSON. Как правило, используется для передачи данных для аутентификации в клиент-серверных приложениях. Токены создаются сервером, подписываются секретным ключом и передаются клиенту, который в дальнейшем использует данный токен для подтверждения своей личности.

Структура JWT:

- header
- payload
- signature

Структура JWT:

Header: Хедер *JWT* содержит информацию о том, как должна вычисляться *JWT* подпись.

header = { "alg": "HS256", "typ": "JWT"}

поле alg, которое определяет алгоритм хеширования. Он будет использоваться при создании подписи. HS256. Это единственное обязательное поле

Структура JWT:

Payload: это полезные данные, которые хранятся внутри *JWT*. Эти данные также называют *JWT-claims* (заявки). Это тоже объект. Существует список стандартных *заявок* для *JWT* payload — вот некоторые из них:

- *iss* (issuer) — определяет приложение, из которого отправляется токен.
- *sub* (subject) — определяет тему токена.
- *exp* (expiration time) — время жизни токена.

Структура JWT:

Signature: вычисляется на основании headers и payload и зависит от выбранного алгоритма

Токены могут быть перекодированы в компактное представление (JWS/JWE Compact Serialization): к заголовку и полезной нагрузке применяется алгоритм кодирования Base64-URL, после чего добавляется подпись и все три элемента разделяются точками («.»).

JWT:

eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJ0b2tlbl90eXBlIjoicmVmcmVzaCIsImV4cCI6MTY0MDY0NDUzMiwianRpIjoiYTkyYzUxNGJkMzYxNGQ1OTlmYmNhZWFiZjMxYzUyZDciLCJ1c2VyX2lkIjo3fQ.BIdG19rRnVB0XRKbb5XuT3pmwaB__35tEm5gcOehxmc

HEADER:

```
{
  "typ": "JWT",
  "alg": "HS256"
}
```

PAYLOAD:

```
{
  "token_type": "refresh",
  "exp": 1640644532,
  "jti": "a92c514bd3614d599fbcaeabf31c52d7",
  "user_id": 7
}
```

VERIFY SIGNATURE

```
HMACSHA256(
  base64UrlEncode(header) + "." +
  base64UrlEncode(payload),
  your-256-bit-secret
) ☐ secret base64 encoded
```