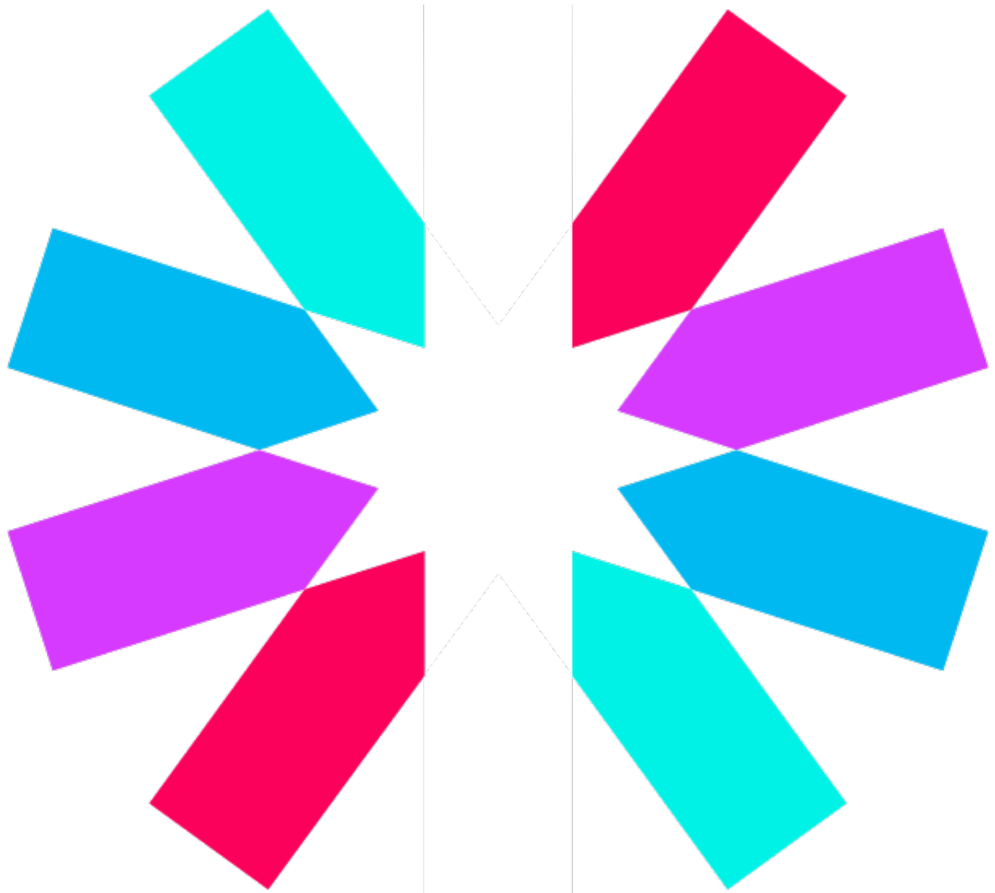
REST Authentication with JWT



🇦🇷 Nacho Anaya

@ianaya89

- Full Stack Developer, Tech Trainer & Speaker
- Ambassador @Auth0
- Organizer @Vuenos_Aires

🤔 Why token authentication?

🤔 Why token authentication?

> Stateless

🤔 Why token authentication?

> Decoupled

🤔 Why token authentication?

> Scalable

🤔 Why JWT?

🤔 Why JWT?

> Standard RFC 7519

🤔 Why JWT?

> Self Contained

🤔 Why JWT?

> Compact

🤔 Why JWT?

> Signed

HMAC - RSA - ECDSA

🤔 Why JWT?

> JSON 🙌

🤔 What is JWT?

🤔 What is JWT?

header.payload.signature

+ Base64

🤔 What is JWT?

```
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.
eyJzdWIiOiIxMjM0NTY3ODkwIiwibmFtZSI6IkpvaG4
gRG9lIiwiaXNTb2NpYWwiOnRydWV9.
4pcPyMD09olPSyXnrXCjTwXyr4BsezdI1AVTmud2fU4
```

🤦‍♀️ Header

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

👌 Payload

```
{
  "id": "1234567890",
  "name": "John Doe",
  "admin": true,
  "iss": "https://api.com",
  "exp": 1510745797148
}
```

👌 Payload

```
{
  "id": "1234567890",
  "name": "John Doe",
  "admin": true,
  "iss": "https://api.com",
  "exp": 1510745797148
}
```

✍️ Signature

```
const data = base64urlEncode( header ) + '.' +
  base64urlEncode( payload )

HMACSHA256(data, 'your_secret_message')
```

✍️ Signature

```
const data = base64urlEncode( header ) + '.' +
  base64urlEncode( payload )

HMACSHA256(data, 'your_secret_message')
```

ALGORITHM HS256

Encoded PASTE A TOKEN HERE

```
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWIiOiIxMjM0NTY3ODkiLCJuYW1lIjoiSm9obiBEb2UiLCJhZG1pbiI6dHJ1ZX0.R1KYYj1fOLNw_-B14ggJX8ohXj1E8iqBPEdKQf-mFas
```

Decoded EDIT THE PAYLOAD AND SECRET (ONLY HS256 SUPPORTED)

HEADER: ALGORITHM & TOKEN TYPE

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

PAYLOAD: DATA

```
{
  "sub": "123456789",
  "name": "John Doe",
  "admin": true
}
```

VERIFY SIGNATURE

```
HMACSHA256(
  base64UrlEncode(header) + "." +
  base64UrlEncode(payload),
  secret
) secret base64 encoded
```

🤔 When to use it?

🤔 When to use it?

> Authentication

> Information Exchange

🤔 Where to use it?

🤔 Where to use it?

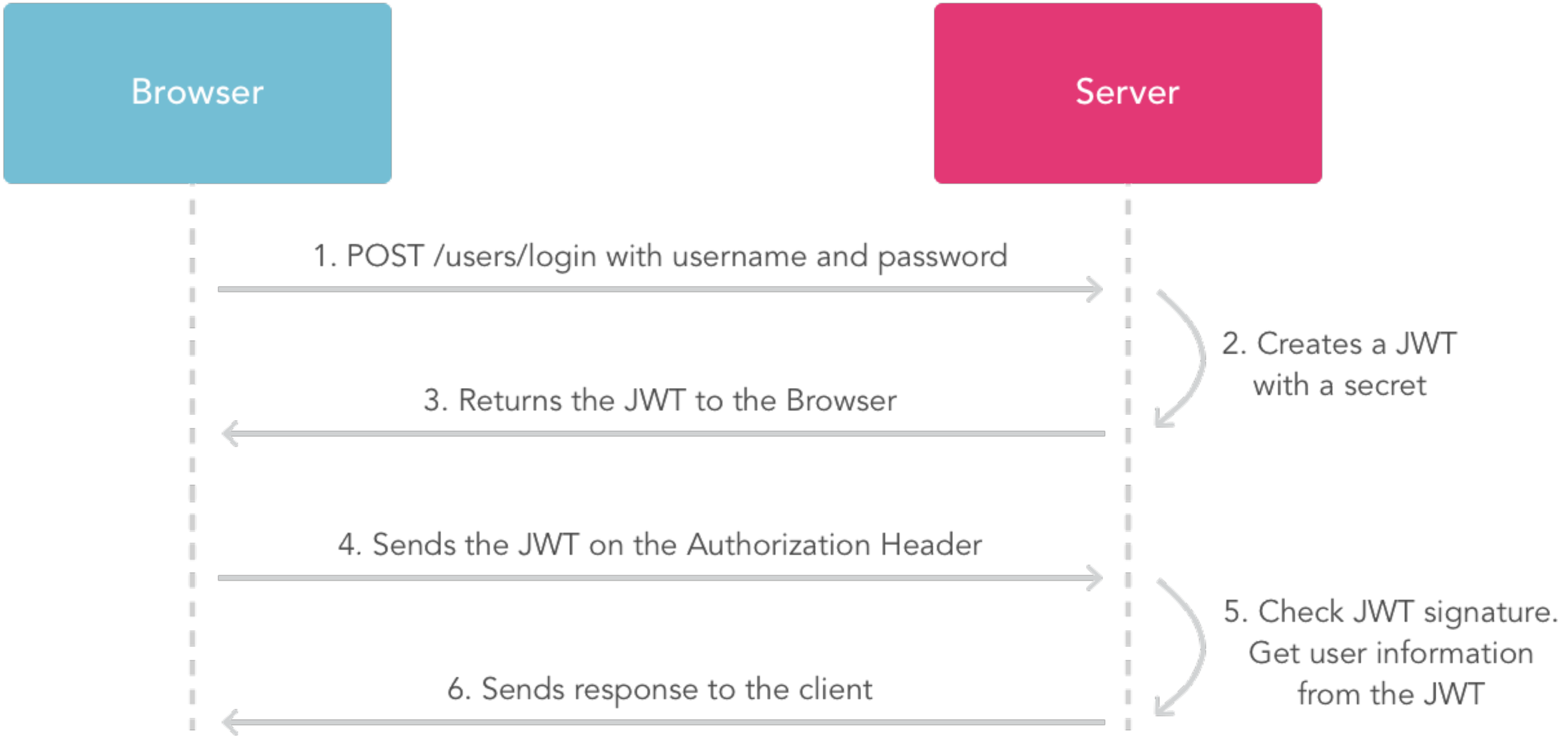
SPA's - Mobile

Serverless - IoT


...and?

REST API's

🤔 How does it work with REST?


Browser
Server
1. POST /users/login with username and password
2. Creates a JWT with a secret
3. Returns the JWT to the Browser
4. Sends the JWT on the Authorization Header
5. Check JWT signature. Get user information from the JWT
6. Sends response to the client

🤔 How does it work with REST?

1. Sends Credentials

```
POST /login

{
    "user": "ianaya89",
    "password": "dont-hack-me"
}
```

🤔 How does it work with REST?

2. Creates JWT

```
const jwt = require('jsonwebtoken')

// POST /login
function login (req, res, next) {
  // Validates user credentials...

  const payload = { user: 'ianaya89', role: 'admin' }

  const token = jwt.sign(payload, 'this_is_super_secret')
  res.status(201).send({ token: `Bearer ${token}` })
}

router.post('/login', login)
```

🤔 How does it work with REST?

3. Returns JWT

```
const jwt = require('jsonwebtoken')

// POST /login
function login (req, res, next) {
  // Validates user credentials...

  const payload = { user: 'ianaya89', role: 'admin' }

  const token = jwt.sign(payload, 'this_is_super_secret')
  res.status(201).send({ token: `Bearer ${token}` })
}

router.post('/login', login)
```

🤔 How does it work with REST?

4. Gets a resource

```
GET /resource
Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.
eyJzdWIiOiIxMjM0NTY3ODkiLCJuYW1lIjoiSm9obiBEb2UiLCJhZG1pbiI6ZmFsc2V9.
b99O1RrYbHtWJ3MGZXkdADZkmiLm9HNliRccKxMPDuc
```

🤔 How does it work with REST?

5. Verifies token

```
const jwt = require('jsonwebtoken')

// GET /resource
function getResource (req, res, next) {
  try {
    const payload = jwt.verify(token, 'this_is_super_secret')
  }
  catch (err) {
     return res.sendStatus(401)
  }
}

router.get('/resource', getResource)
```

🤔 How does it work with REST?

6. Sends response

```
const jwt = require('jsonwebtoken')

// GET /resource
function getResource (req, res, next) {
  try {
    const payload = jwt.verify(token, 'this_is_super_secret')
    res.send('👌')
  }
  catch (err) {
     return res.sendStatus(401)
  }
}

router.get('/resource', getResource)
```

🤔 How does it work with REST?

6. Sends response

```
const jwt = require('jsonwebtoken')

// GET /resource
function getResource (req, res, next) {
  try {
    const payload = jwt.verify(token, 'this_is_super_secret')

    if (payload.role !== 'admin') {
      return res.sendStatus(403)
    }

    res.send('👌')
  }
  catch (err) {
     return res.sendStatus(401)
  }
}

router.get('/resource', getResource)
```

🤔 Which languages are supported?

🤔 Which languages are supported?

> "All" of them

Libraries

🤔 Is JWT secure?

🤔 Is JWT secure?

👌 Yes

🤔 Is JWT secure?

🤫 But...


MOMMA ALWAYS SAID
"SECURE IS AS SECURE DOES"
HD
memegenerator.net

🤔 Is JWT secure?

> Anyone can view the content

🤔 Is JWT secure?

> No one can modify it

🤔 Is JWT secure?

> JWT is signed not ecnrpyted

🤔 Is JWT secure?

> Keep your "secret" secret 😏

📚 Resources

- jwt.io
- jwt-handbook
- demo-auth-jwt-api

👏 Thanks!

@ianaya89

bit.ly/rest-auth-jwt