JSON Web Token (JWT)

Luděk Benedík

What is JWT

- Open standard (RFC 7519)
- Compact
- Self-contained
- Securely transmitting information
- Digitally signed

Usage of JWT

- Authentication
- Information Exchange

Structure of JWT

- Three parts separated by dots
- Header
- Payload
- Signature
- **aaaaa.bbbbb.ccccc**

JWT Header

```
{
    “typ”: “JWT”,
    “alg”: “RS256”
}
```

Base64Url `eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiJ9`

JWT Payload

- Reserved claims
  - **iss** (issuer)
  - **exp** (expiration time)
  - **sub** (subject)
  - ...
- Public claims
- Private claims

JWT Payload

```
{
    “sub”: 123456789,
    “admin”: true
}
```

Base64Url `eyJzdWIiOjEyMzQ1Njc4OSwiYWRtaW4iOnRydWV9`

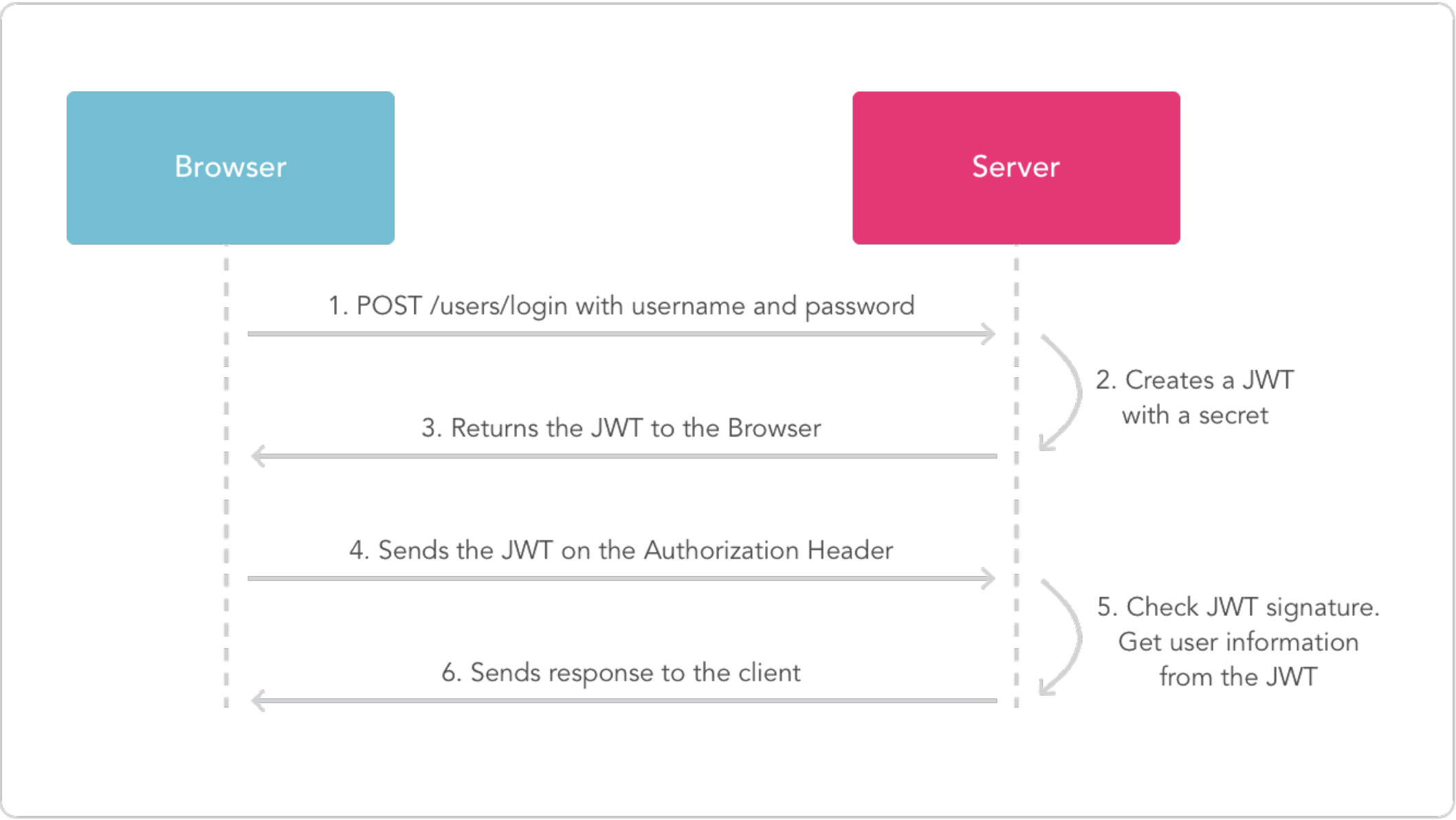
JWT Signature

```
$header          = [‘typ’ => ‘JWT’, ‘alg’ => ‘RS256’];
$headerJson      = json_encode($header);
$headerBase64Url = base64url_encode($headerJson);

$payload          = ['sub' => 123456789, 'admin' => true];
$payloadJson      = json_encode($payload);
$payloadBase64Url = base64url_encode($payloadJson);

$data               = $headerBase64Url . ‘.’ . $payloadBase64Url;
$signature          = rs256($data, $key, $passphrase);
$signatureBase64Url = base64url_encode($signature);

$jwt = $data . ’.’ . $signatureBase64Url;

echo $jwt; // aaaaa.bbbbb.ccccc
```

Receive JWT

- Verify that the JWT contains three dots
- Split JWT to header, payload and signature
- Encode header
- Get algorithm
- Check signature
- Encode payload
- Use data from payload

Authentication



Information Exchange

- Public/Private keys pairs
- Create → sign (private key) → send
- Receive → verify (public key) → Use/Refuse

Documentation

- https://jwt.io/
- https://tools.ietf.org/html/rfc7519

Composer libraries

- firebase/php-jwt
- namshi/jose
- lcobucci/jwt
- emarref/jwt
- spomky-labs/jose
- gree/jose

Questions?