



# Cloud-Native Security

<https://github.com/andifalk/cloud-security-workshop>

Andreas Falk

# Vorstellung

---

Andreas Falk  
Novatec Consulting



[andreas.falk@novatec-gmbh.de](mailto:andreas.falk@novatec-gmbh.de) / [@andifalk](https://www.novatec-gmbh.de/beratung)

<https://www.novatec-gmbh.de/beratung/agile-security>

# Agenda

---

1. Intro:
  - Why Security?
  - What's Cloud-Native Security?
2. OWASP & OWASP Top 10
3. Client Security
4. Server Security
5. Security in the Cloud
6. Practice



---

# Why Security ?

#securityfails

# Cayla Doll - The Spy Toy



Troy Hunt [@troyhunt](#) Folge ich

Germany not mucking around with spying toys "Germany Issues Kill Order for a Domestic Spy—Cayla the Toy Doll"

Data from connected CloudPets teddy bears leaked and ransomed, exposing kids' voice messages

28 FEBRUARY 2017

[Germany Issues Kill Order for a Domestic Spy—Cayla the Toy Doll](#)  
On a campaign to promote digital privacy, authorities warn that "My Friend Cayla" makes children vulnerable to malicious surveillance. They've ordered parents to d...  
[wsj.com](#)

# Ashley Madison - Serious Life Changer



Troy Hunt @troyhunt · 10. Dez.

Good insight into the human impact: "Scared, dead, relieved: How the **Ashley Madison** hack changed its victims' lives" [fusion.net/story/242502/a...](https://fusion.net/story/242502/a...)



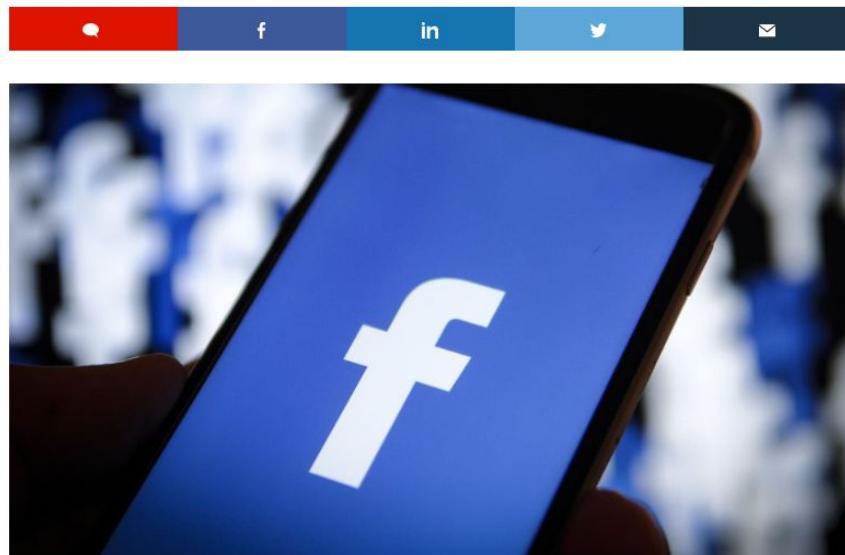
# Facebook Leak - Big Data Exposure

## Over 540 million Facebook records found on exposed AWS servers

Leak originated at two third-party companies that had collected Facebook data on their own servers.



By Catalin Cimpanu for Zero Day | April 3, 2019 -- 18:32 GMT (19:32 BST) | Topic: Security



### MORE FROM CATALIN CIMANU



Windows 10  
Microsoft changes how Windows 10 disconnects USB storage devices



Developer  
After Chrome, Firefox will also support off-screen image lazy loading



Security  
FBI criticized for delaying breach notifications, including insufficient details



Security  
IoT botnet targeting your enterprise? Nope. Just a kid with an ExploitDB account

### NEWSLETTERS

#### ZDNet Security

Your weekly update on security around the globe, featuring research, threats, and more.

# EasyJet Hacked For Four Months, Data On 9 Million Customers And 2,000 Credit Cards Stolen



**Thomas Brewster** Forbes Staff

Cybersecurity

*Associate editor at Forbes, covering cybercrime, privacy, security and surveillance.*



# Crypto Mining Via K8s Dashboard

Source: <https://blog.heptio.com>

## On Securing the Kubernetes Dashboard



Joe Beda [Follow](#)

Feb 28, 2018 · 13 min read

Recently Tesla (the car company) was alerted, by security firm RedLock, that their Kubernetes infrastructure was compromised. The attackers were using Tesla's infrastructure resources to mine cryptocurrency. This type of attack has been called “cryptojacking”.

The vector of attack in this case was a Kubernetes Dashboard that was exposed to the general internet with no authentication and elevated privileges. Not only this, but core AWS API keys and secrets were visible. How do you prevent this from happening to you?

# Shodan.io - Google for “Hackers”



The search engine for **Security**

Shodan is the world's first search engine for Internet-connected devices.

Create a Free Account      Getting Started

**Explore the Internet of Things**

Use Shodan to discover which of your devices are connected to the Internet, where they are located and who is using them.

**Monitor Network Security**

Keep track of all the computers on your network that are directly accessible from the Internet. Shodan lets you understand your digital footprint.

**See the Big Picture**

Websites are just one part of the Internet. There are power plants, Smart TVs, refrigerators and much more that can be found with Shodan!

**Get a Competitive Advantage**

Who is using your product? Where are they located? Use Shodan to perform empirical market intelligence.

# haveibeenpwned.com - Exposed Accounts & Passwords

';--have i been pwned?

Check if you have an account that has been compromised in a data breach

## Pwned Passwords

email address

Pwned Passwords are 555,278,657 real world passwords previously exposed in data breaches. This exposure makes them unsuitable for ongoing use as they're at much greater risk of being used to take over other accounts. They're searchable online below as well as being downloadable for use in other online systems. [Read more about how HIBP protects the privacy of searched passwords.](#)

password



pwned?

# Pay attention on what you copy from stackoverflow.com

---

This is somewhat simple

```
string inp = "hai";
StringBuilder strb = new StringBuilder();
foreach (char s in inp)
{
    int sin = s + 5;
    char newch = (char)sin;
    strb.Append(newch);
}
string output = strb.ToString();
```

Now the output contains the encrypted string "mfn" (ie., 5 letters away from the original )in it....

# Security is a really comprehensive topic

---

SQLInjection CSRF XSS OWASP OAuth2

OpenID-Connect AbUser-Stories

Authentication Authorization Secure Coding

Security-Testing SSO DoS Sensitive-Data

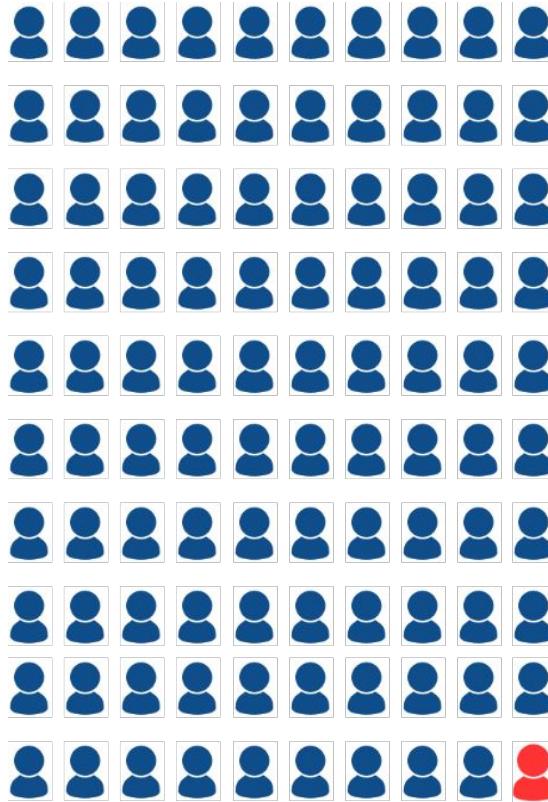
Data-Privacy Crypto Code-Reviews Threat-

Modeling Architecture Dependencies DAST

SAML SAST DevSecOps

# 1 Security-Professional for 100 Developers

---

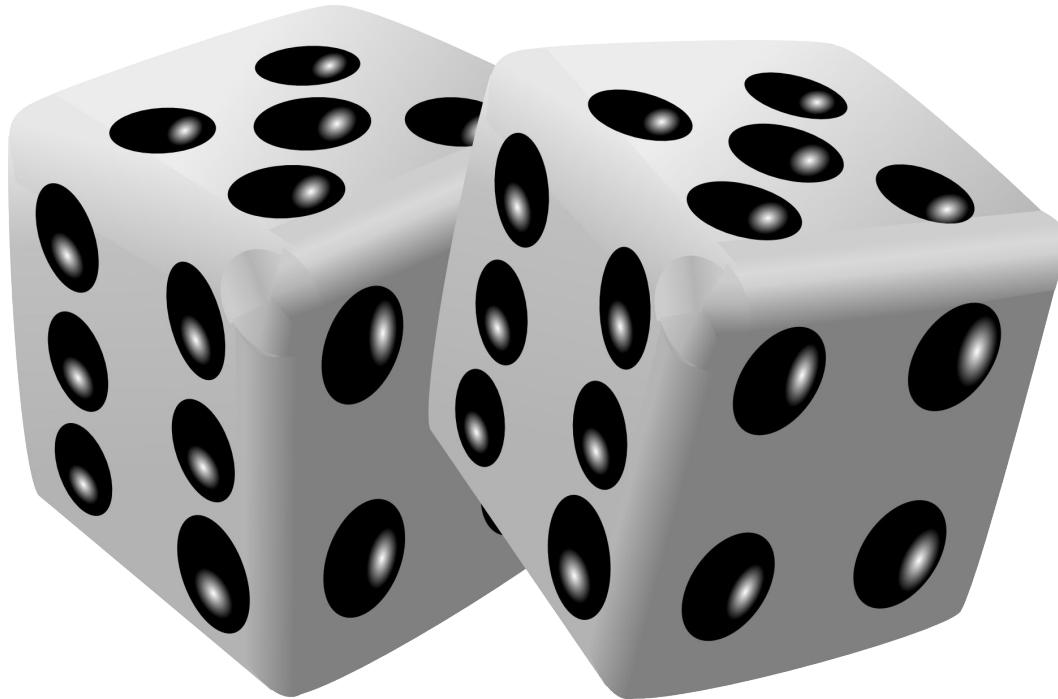


Source:

<https://www.sonatype.com/devops-survey-report>

# Security as a game of chance?

---



---

# **Open Web Application Security Project**

<https://www.owasp.org>

# OWASP Top 10 - 2017



OWASP Top 10 - 2013	→	OWASP Top 10 - 2017
A1 – Injection	→	A1:2017-Injection
A2 – Broken Authentication and Session Management	→	A2:2017-Broken Authentication
A3 – Cross-Site Scripting (XSS)	↳	A3:2017-Sensitive Data Exposure
A4 – Insecure Direct Object References [Merged+A7]	U	A4:2017-XML External Entities (XXE) [NEW]
A5 – Security Misconfiguration	↳	A5:2017-Broken Access Control [Merged]
A6 – Sensitive Data Exposure	↗	A6:2017-Security Misconfiguration
A7 – Missing Function Level Access Contr [Merged+A4]	U	A7:2017-Cross-Site Scripting (XSS)
A8 – Cross-Site Request Forgery (CSRF)	☒	A8:2017-Insecure Deserialization [NEW, Community]
A9 – Using Components with Known Vulnerabilities	→	A9:2017-Using Components with Known Vulnerabilities
A10 – Unvalidated Redirects and Forwards	☒	A10:2017-Insufficient Logging&Monitoring [NEW,Comm.]

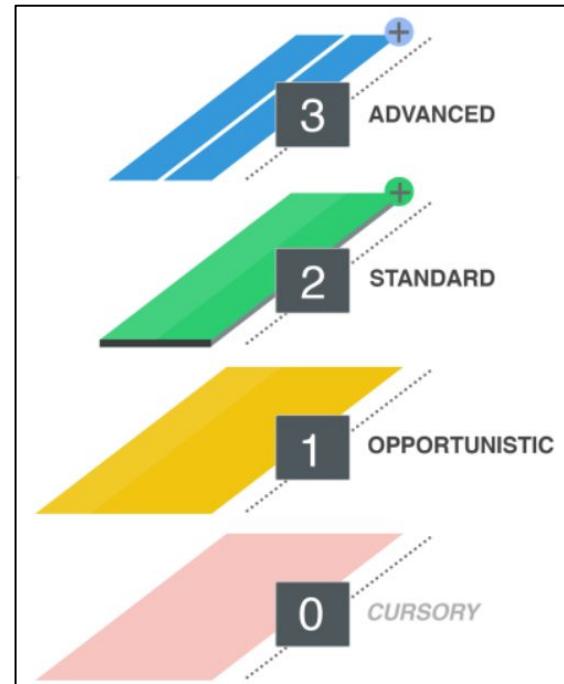
## Other OWASP Offerings

---



[https://www.owasp.org/index.php/OWASP\\_Proactive\\_Controls](https://www.owasp.org/index.php/OWASP_Proactive_Controls)

## Application Security Verification Standard



<https://github.com/OWASP/ASVS>

# OWASP Zap - Open Source Penetration Testing Tool

The screenshot shows the main interface of the OWASP ZAP web application. At the top, there is a navigation bar with icons for 'Quick Start' (a lightning bolt), 'Request' (a green arrow pointing right), 'Response' (a green arrow pointing left), and a '+' sign. Below the navigation bar, the title 'Welcome to OWASP ZAP' is displayed in large, bold letters. A sub-header below the title reads: 'ZAP is an easy to use integrated penetration testing tool for finding vulnerabilities in web applications.' Another line of text says: 'If you are new to ZAP then it is best to start with one of the options below.' Below this text are three large buttons: 'Automated Scan' (with a lightning bolt icon), 'Manual Explore' (with a lightning bolt and target icon), and 'Learn More' (with a question mark icon). At the bottom of the page, there is a 'News' section containing a message: 'ZAP 2.8.0 includes an innovative 'Heads Up Display' (HUD)' followed by a 'Learn More' button and a red 'X' icon.

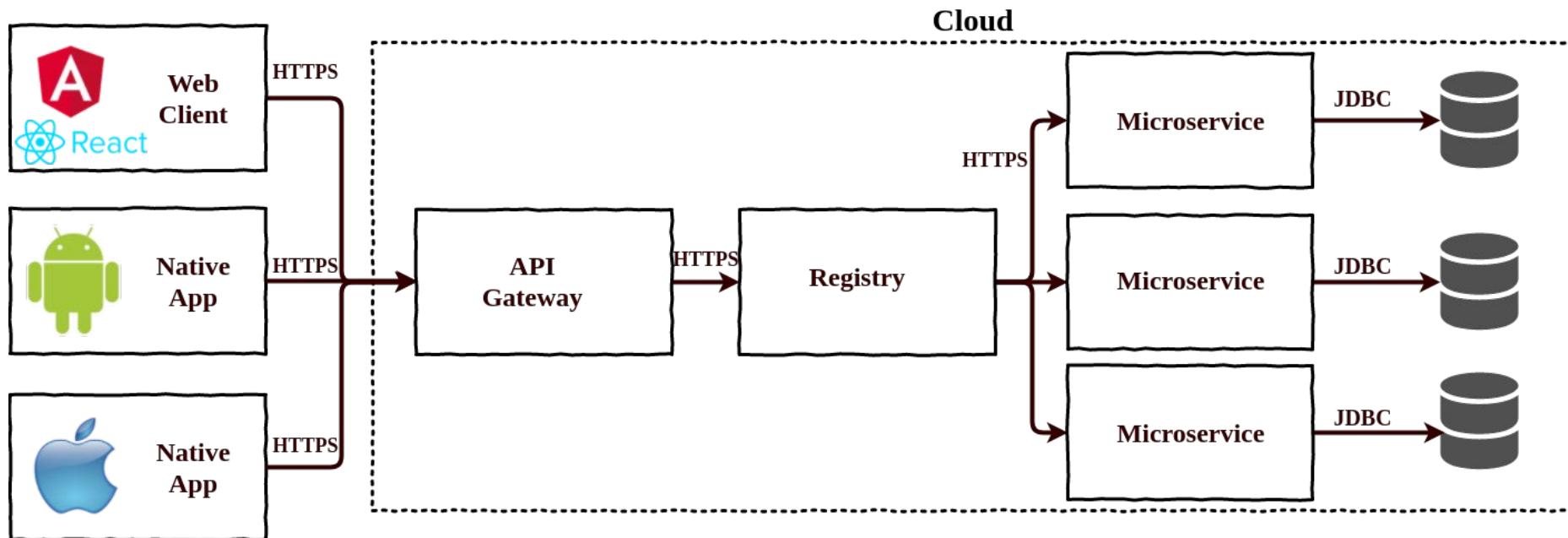
<https://www.zaproxy.org/>

---

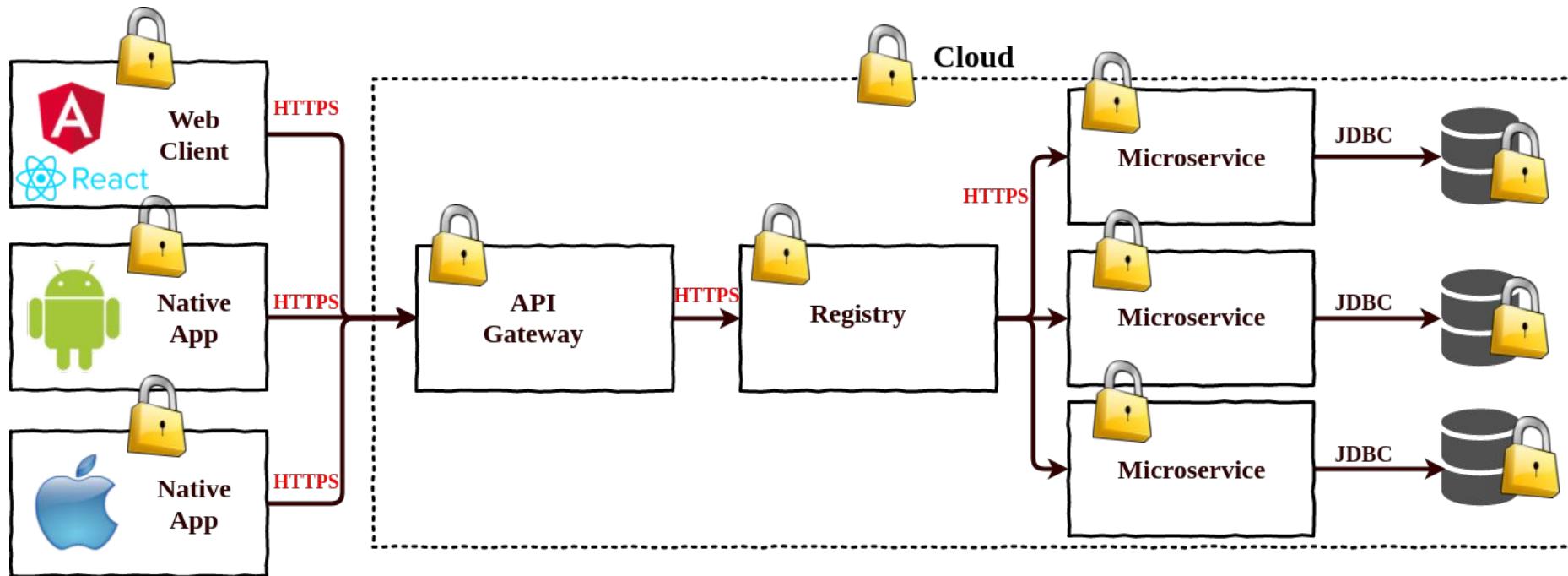
# Cloud-Native Security

Cloud and Platforms (CloudFoundry / Kubernetes)

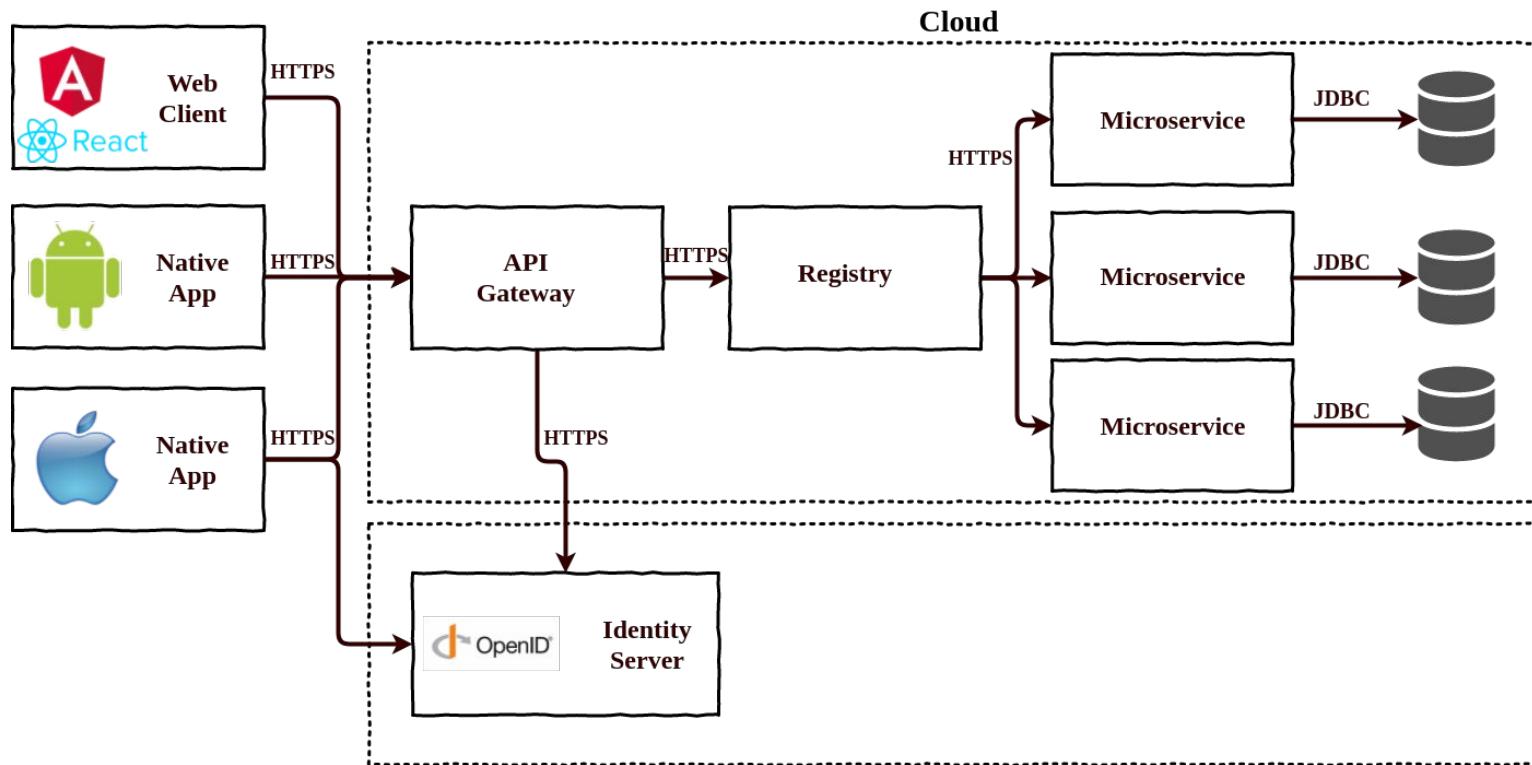
# Cloud-Native Architecture



# Secure Cloud-Native Architecture



# Cloud-Native Architecture with Identity-Management



# Technology Stacks

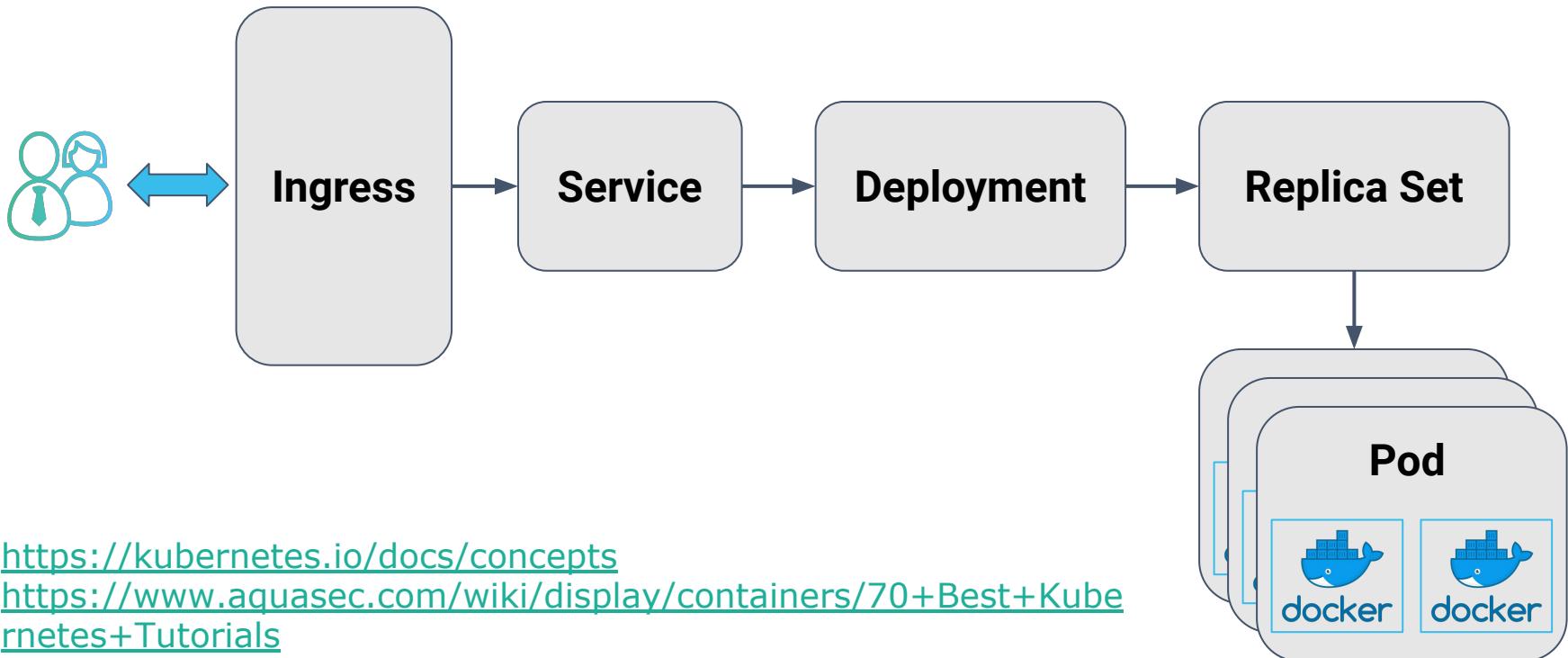
---



Applications



# Kubernetes Basics

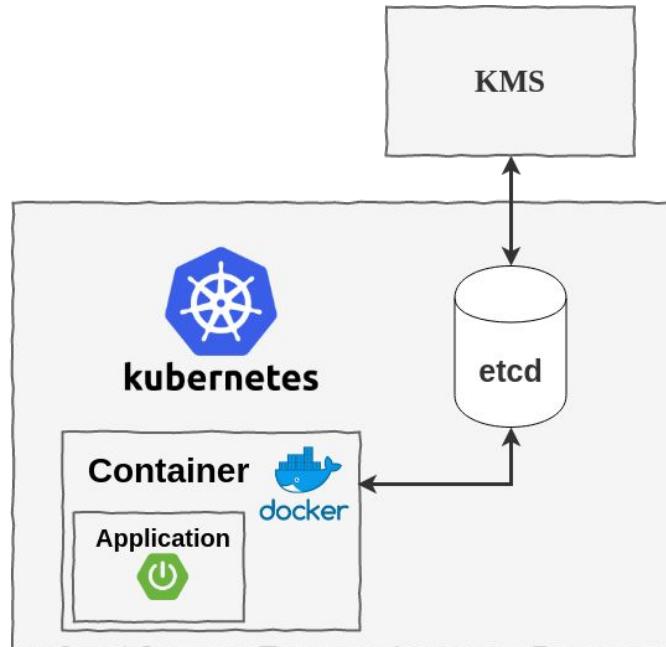
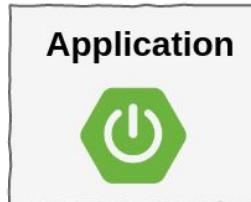


<https://kubernetes.io/docs/concepts>

<https://www.aquasec.com/wiki/display/containers/70+Best+Kubernetes+Tutorials>

# Cloud-Native Security in Kubernetes

Sensitive Data  
Exposure    Dynamic Analysis  
Static Analysis    Dependency Check  
Authentication    Authorization



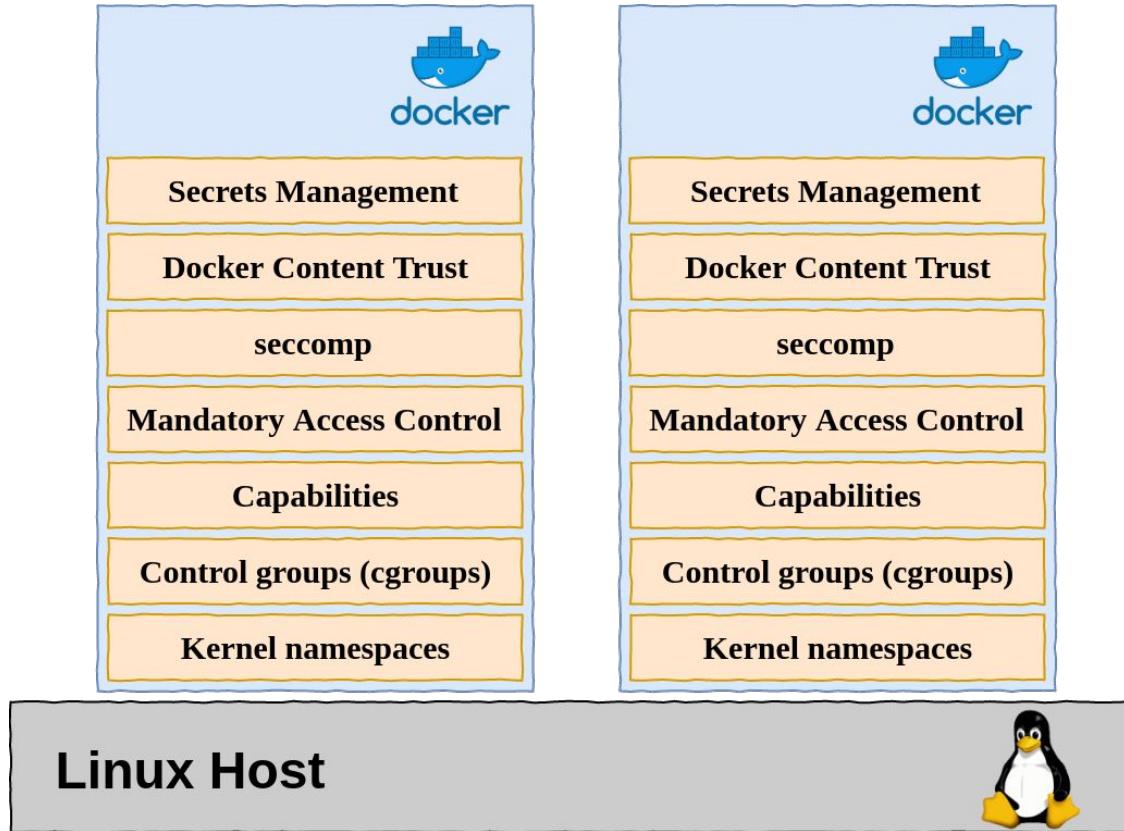
Application Security

Container Security

Kubernetes Security

Kubernetes Secrets

# Docker Security Basics



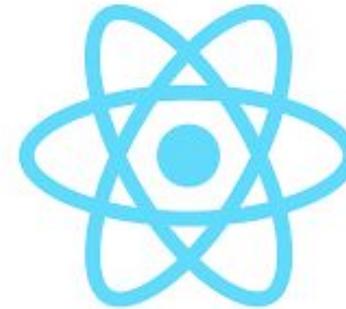
---

# Client Side Security

Frontends

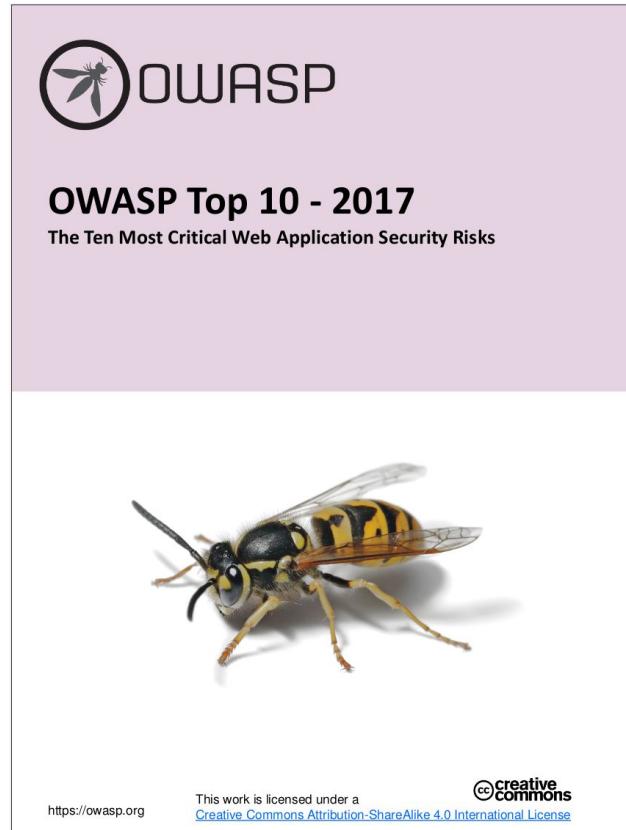
# Cloud Side Security (Single Page Applications)

---



<https://angular.io>  
<https://vuejs.org>  
<https://reactjs.org>

# OWASP Top A7: Cross-Site Scripting (XSS)



# Cross-Site Scripting (XSS)

---

- Injection of malicious scripts into web sites
  - Reflected XSS
  - Persistent XSS
  - DOM based XSS

```
<script>alert(123)</script>  
"><script>alert(document.cookie)</script>
```

# Cross-Site Scripting Defenses

---

- Output Escaping (Angular, React.js, Vue.js, JSF, Thymeleaf)
- Sanitizing (Angular, DOMPurify)
- Protect Sessions (e.g. in web.xml)

```
<session-config>
    <cookie-config>
        <http-only>true</http-only>
        <secure>true</secure>
    </cookie-config>
    <tracking-mode>COOKIE</tracking-mode>
</session-config>
```



---

# Angular

Angular.js = “Legacy” Angular 1.x versions

Angular = Current Angular version (latest is V8)

# Angular Security

---

“...The basic idea is to implement automatic secure escaping for all values that can reach the DOM...  
By default, with no specific action for developers, Angular apps must be secure...”

<https://github.com/angular/angular/issues/8511>

# Angular XSS Protection

---

ANGULAR TEMPLATE = **SAFE**

INPUT VALUES = **UNSAFE**

# Angular Component (TypeScript)

---

```
@Component({
  selector: 'app-root',
  templateUrl: 'app.component.html',
  styleUrls: ['app.component.css']
})
export class AppComponent {
  untrustedHtml:string =
    '<em><script>alert("hello")</script></em>';
}
```

# Angular Template

---

## HTML BINDINGS:

...

```
<h3>Encoded HTML snippet</h3>
```

```
<h3 class="trusted">{{untrustedHtml}}</h3>
```

...

```
<h3>Sanitized HTML snippet</h3>
```

```
<h3 class="trusted" [innerHTML]="untrustedHtml"></h3>
```

...

# Unsafe Angular APIs

---

## API List

Type: All

Status: Security Risk

- ElementRef:  
Direct access to DOM!
- DomSanitizer:  
Deactivates XSS-Protection!

**Do NOT use!**

<https://angular.io/docs>

core

c ElementRef

platform-browser

c DomSanitizer

# Security Comparison Angular vs. React

Source: <https://snyk.io/blog/javascript-frameworks-security-report-2019>

Item	Angular	React
Security page	✓ <a href="https://angular.io/guide/security">https://angular.io/guide/security</a>	✗ React's website ( <a href="https://reactjs.org">https://reactjs.org</a> ) does not mention any security guidelines, except for the dangerouslySetInnerHTML function reference in the DOM Elements section of the API Reference documentation.
Security contact	✓ <a href="mailto:security@angular.io">security@angular.io</a>	✗ No security contact
Responsible disclosure policy	✓ Backed by the internal security teams at Google and based on Google security philosophy. Reference: <a href="https://www.google.com/about/appsecurity/">https://www.google.com/about/appsecurity/</a>	✗ No responsible disclosure policy
Examples of vulnerable projects	✓ <a href="https://angular.io/generated/live-examples/security/stackblitz.html">https://angular.io/generated/live-examples/security/stackblitz.html</a>	✗ No references to any examples of vulnerable projects
Built-in sanitization	✓ DomSanitizer provides a built-in sanitization function for untrusted values. Reference: <a href="https://angular.io/api/platform-browser/DomSanitizer#sanitize">https://angular.io/api/platform-browser/DomSanitizer#sanitize</a>	✗ Potentially malicious input sanitization is at the users' discretion to be implemented via 3rd-party libraries, such as DOMPurify. Reference: <a href="https://github.com/cure53/DOMPurify">https://github.com/cure53/DOMPurify</a>
Content Security Policy (CSP)	✓ CSP compatibility for Angular v1.x directives. Reference: <a href="https://docs.angularjs.org/api/ng/directive/ngCsp">https://docs.angularjs.org/api/ng/directive/ngCsp</a>	ⓘ Not relevant for React
Cross-Site Request Forgery (CSRF)	✓ CSRF built-in support through Angular's HttpClient service. Reference: <a href="https://angular.io/guide/http">https://angular.io/guide/http</a> and <a href="https://docs.angularjs.org/api/ng/service/\$http">https://docs.angularjs.org/api/ng/service/\$http</a>	ⓘ Not relevant for React as a view library. This is up to the developers to handle using custom code or community modules.

---

# Server Side Security

Java/Kotlin Backends

# Java / Kotlin Server-Side Applications

---



# OWASP Top 10 on Server-Side

---

A1: Injection

A2: Broken Authentication

A3: Sensitive Data Exposure

A5: Broken Access Control

A6: Security Misconfiguration

A9: Using Components With Known  
Vulnerabilities



OWASP

**OWASP Top 10 - 2017**

The Ten Most Critical Web Application Security Risks



<https://owasp.org>

This work is licensed under a  
[Creative Commons Attribution-ShareAlike 4.0 International License](#)



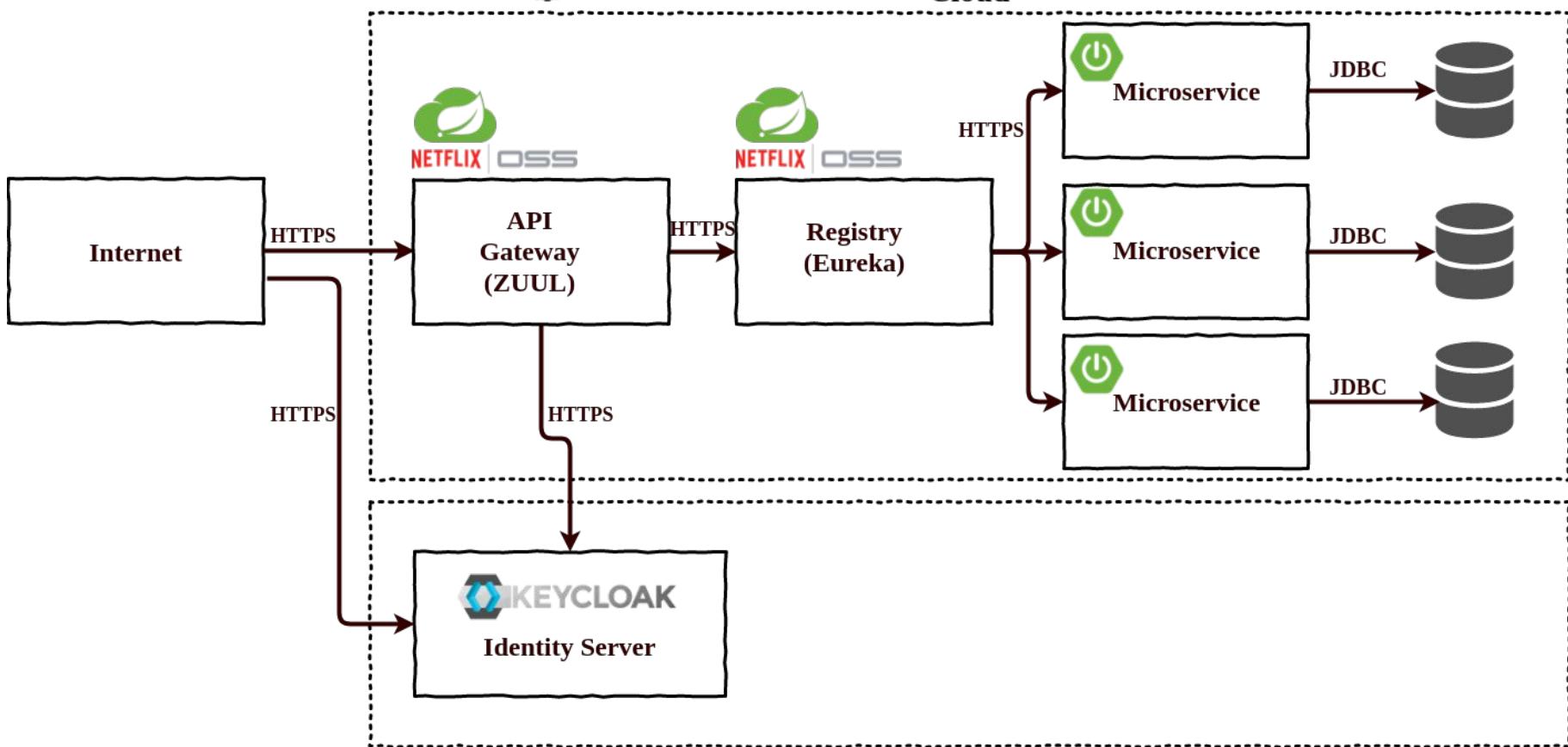
# The Spring Platform

---



<https://spring.io>

# CLOUD FOUNDRY





**Project**

Maven Project  Gradle Project

**Language**

Java  Kotlin  Groovy

**Spring Boot**

2.4 (SNAPSHOT)  2.3.1 (SNAPSHOT)  2.3.0  2.2.8 (SNAPSHOT)

2.2.7  2.1.15 (SNAPSHOT)  2.1.14

**Project Metadata**

Group: com.example

Artifact: demo

Name: demo

Description: Demo project for Spring Boot

Package name: com.example.demo

Packaging:  Jar  War

Java:  14  11  8

## Dependencies

[ADD DEPENDENCIES... CTRL + B](#)

### Spring Web WEB

Build web, including RESTful, applications using Spring MVC. Uses Apache Tomcat as the default embedded container.

### Spring Security SECURITY

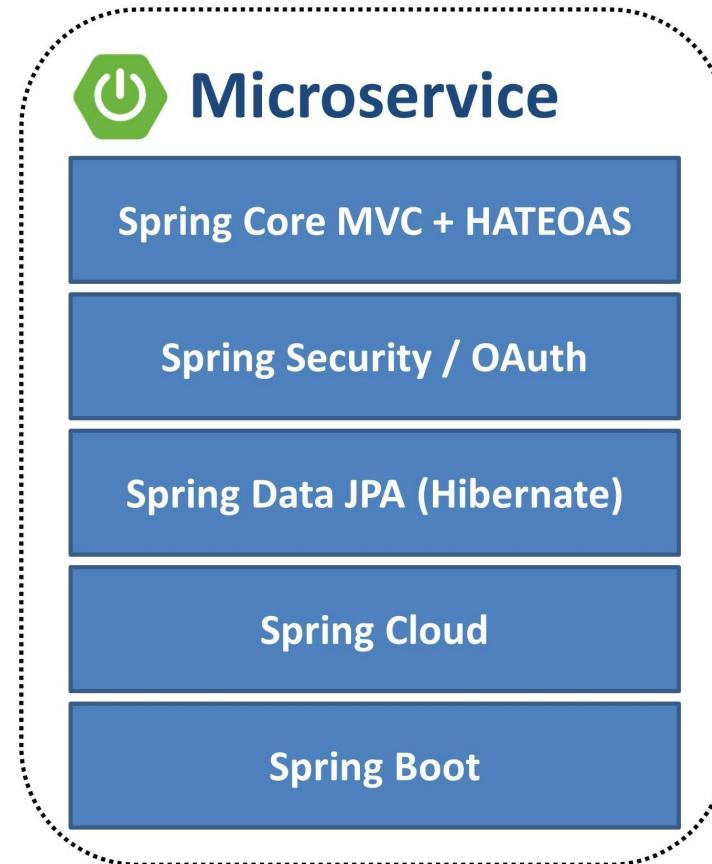
Highly customizable authentication and access-control framework for Spring applications.

### Spring Data JPA SQL

Persist data in SQL stores with Java Persistence API using Spring Data and Hibernate.

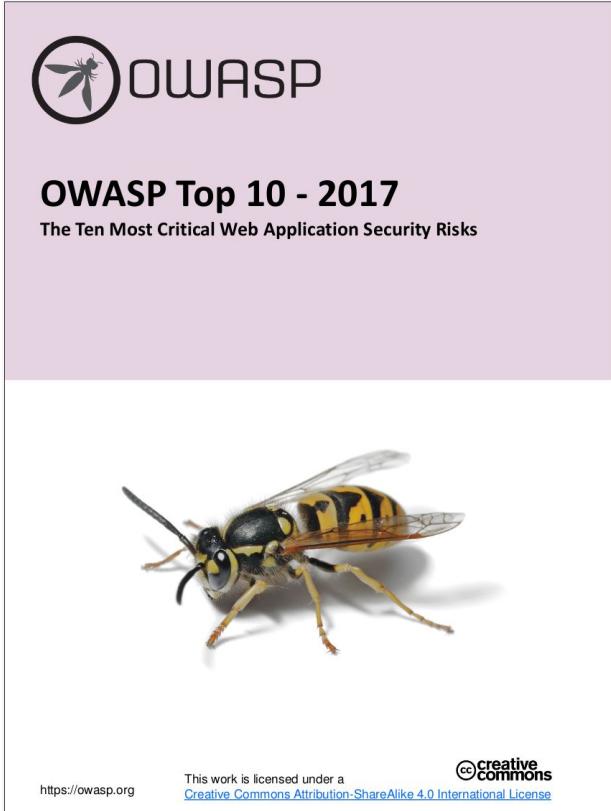
# The Spring Microservice Tech Stack

---



# A2: Broken Authentication

---



# Authentication - Who am I?

---



HTTP 401 -  
UNAUTHORIZED

# Authentication Mechanisms

---

- Basic Authentication / Digest Access Authentication
- Form-based Authentication (i.e. using Session Cookies)
- Client-Certificates (MTLS)
- Kerberos Tickets
- SAML Assertion Tokens
- JSON Web Tokens
- OAuth 2.0 & OpenID Connect
- Proprietary mechanisms like API-Tokens, Siteminder etc.

# Basic Authentication

---

GET / HTTP/1.1

Host: localhost:8080

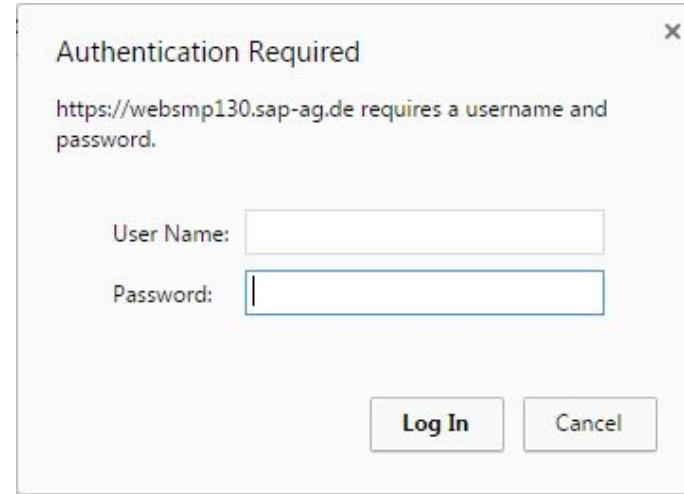
HTTP/1.1 401

WWW-Authenticate: Basic realm="hello"

GET / HTTP/1.1

Host: localhost:8080

Authorization: Basic dXNlcjpzZWNyZXQ=



# Form-Based Authentication

---

POST /login HTTP/1.1

Host: localhost:8080

Content-Type: application/x-www-form-urlencoded

Cookie: JSESSIONID=14965E3A995DA1973F42F308D59727D4

username=user&password=secret&submit=Login

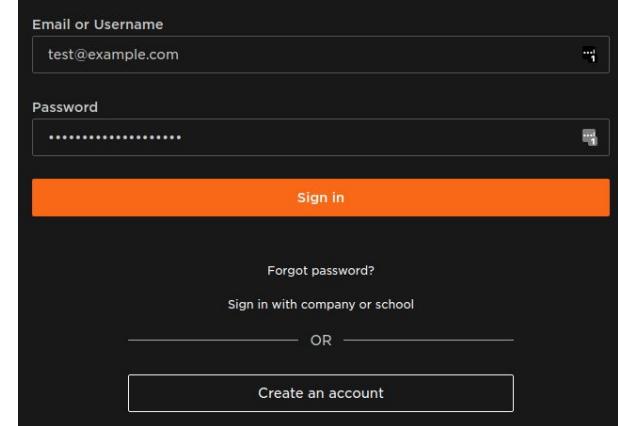
HTTP/1.1 302

Set-Cookie: JSESSIONID=49C632387800316021BE;Path=/; HttpOnly

GET / HTTP/1.1

Host: localhost:8080

Cookie: JSESSIONID=49C632387800316021BE



# Bearer Token Authentication

---

- Used for
  - OAuth 2.0
  - OpenID Connect

GET / HTTP/1.1

Host: localhost:8080

Authorization: Bearer eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiI

# Authentication - Stateful vs. Stateless

---

<b>Session Cookie</b>	<b>Token</b>
With each Request (on same domain)	Manually set as Header
Potential <a href="#">CSRF</a> !	No <a href="#">CSRF</a> possible
One domain	Cross domain ( <a href="#">Cross-Origin Resource Sharing</a> )
Sensitive Info (HTTPS)	Sensitive Info (HTTPS)

# HTTPS (SSL/TLS)

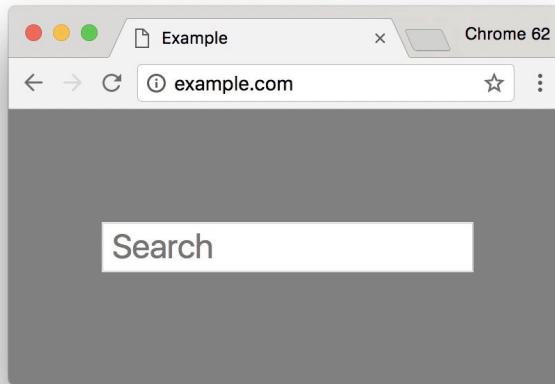
---

- Validation (Destination Server)
- Data Confidentiality (Encryption)
- Data Integrity (Hashing)

# HTTPS (SSL/TLS)

---

- Let's Encrypt
- CloudFlare
- HTTP/2





---

## OAuth 2.0 (RFC 6749)

<https://tools.ietf.org/html/rfc6749>

# Do you like implementing your own Authentication?

Different Clients

Brute Force Prevention

Reset Password Process

Log In

Username or email

Password

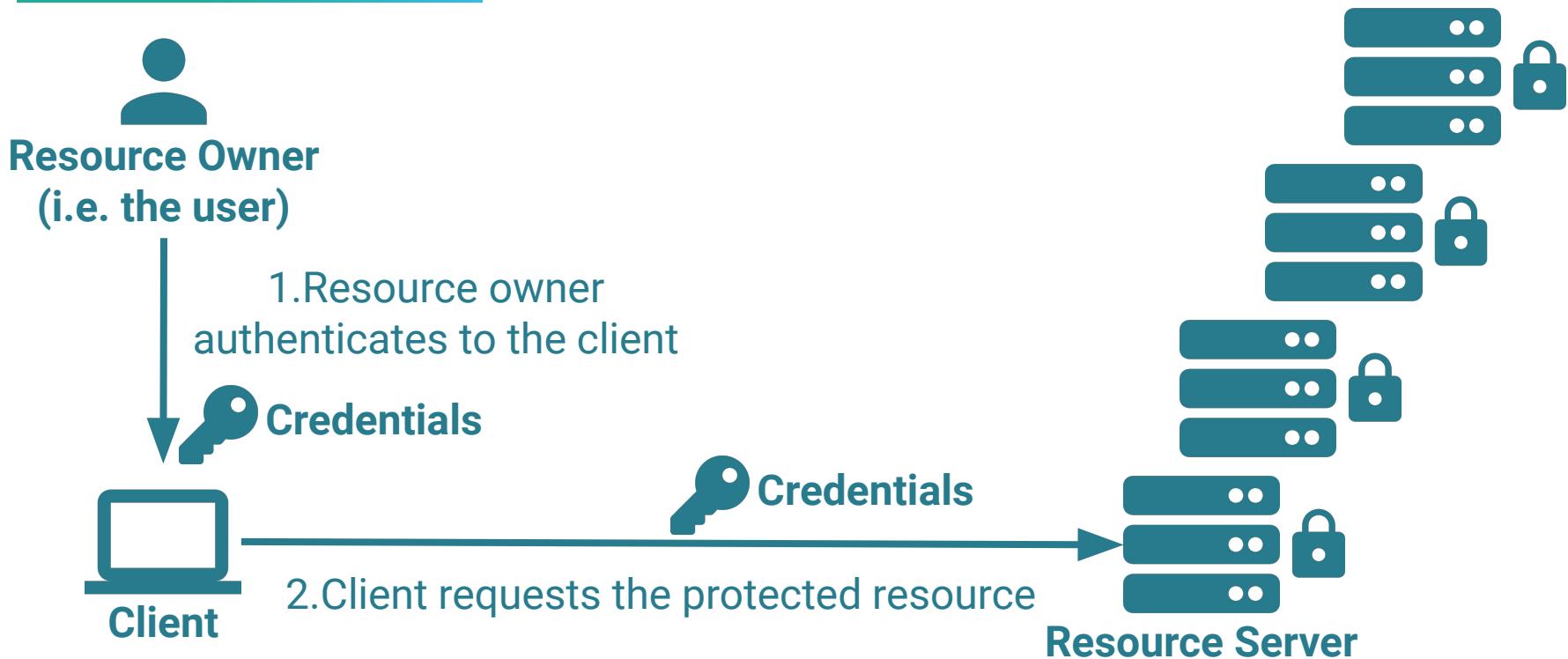
**Log In**

Password Policies

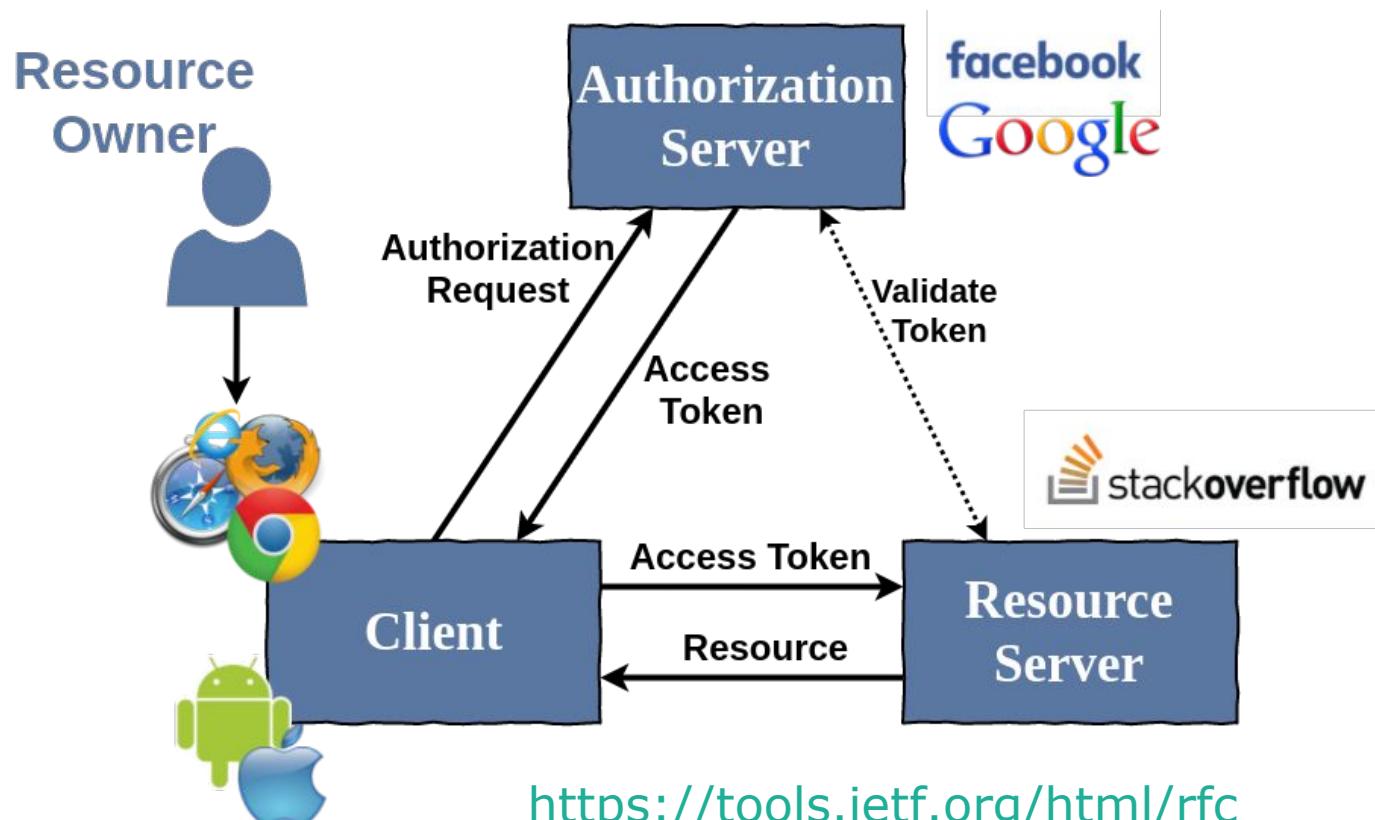
MFA

Secure Password Storage

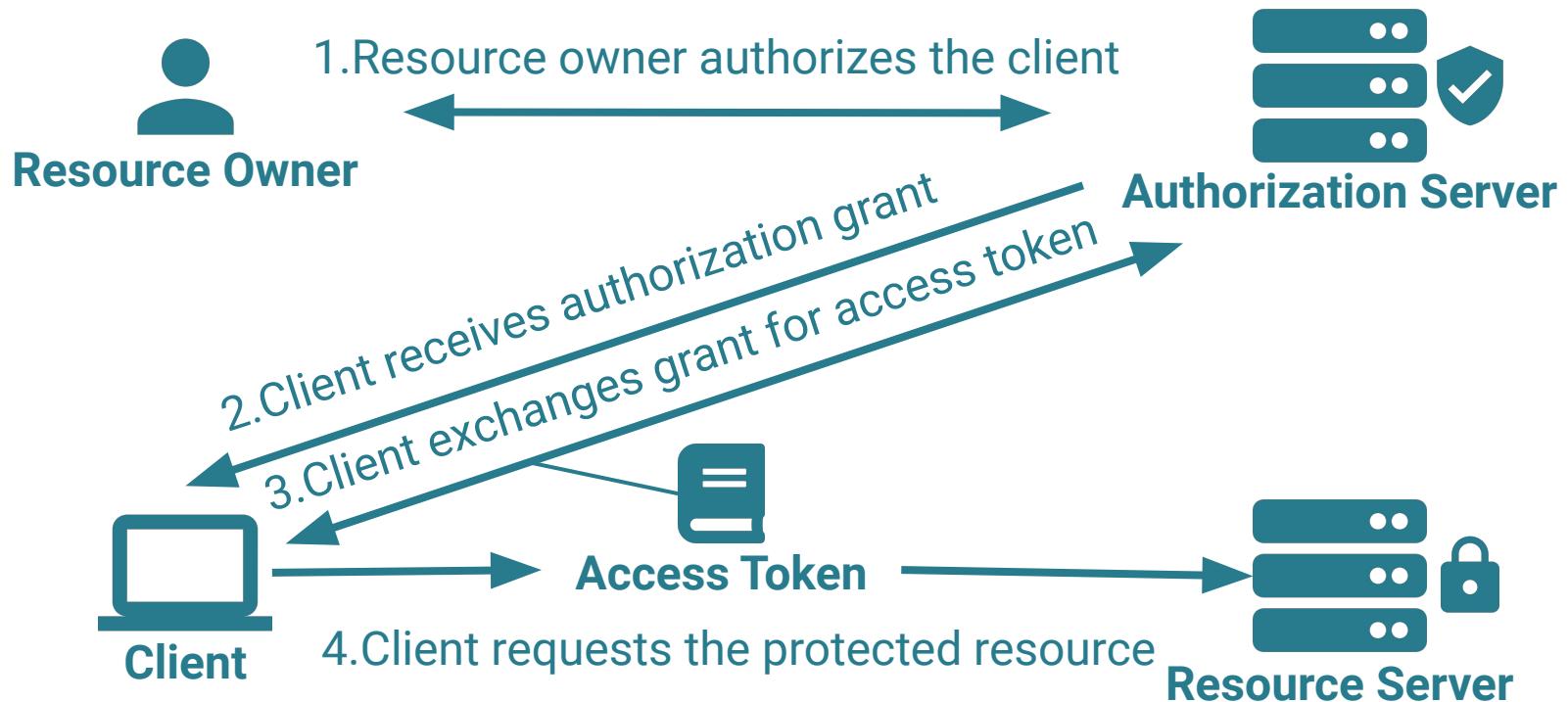
# Spreading Credentials before OAuth 2.0



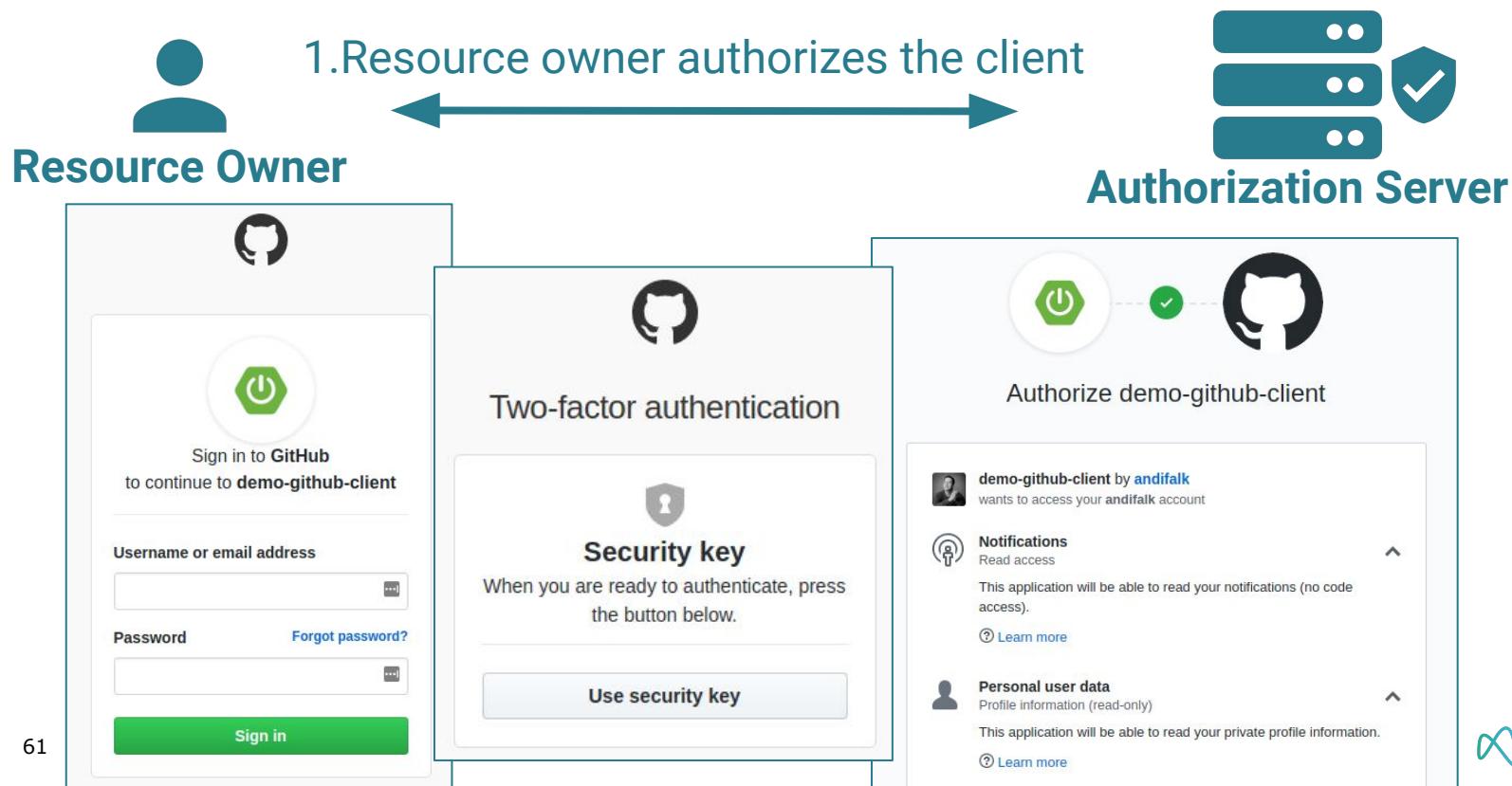
# OAuth 2.0 Roles



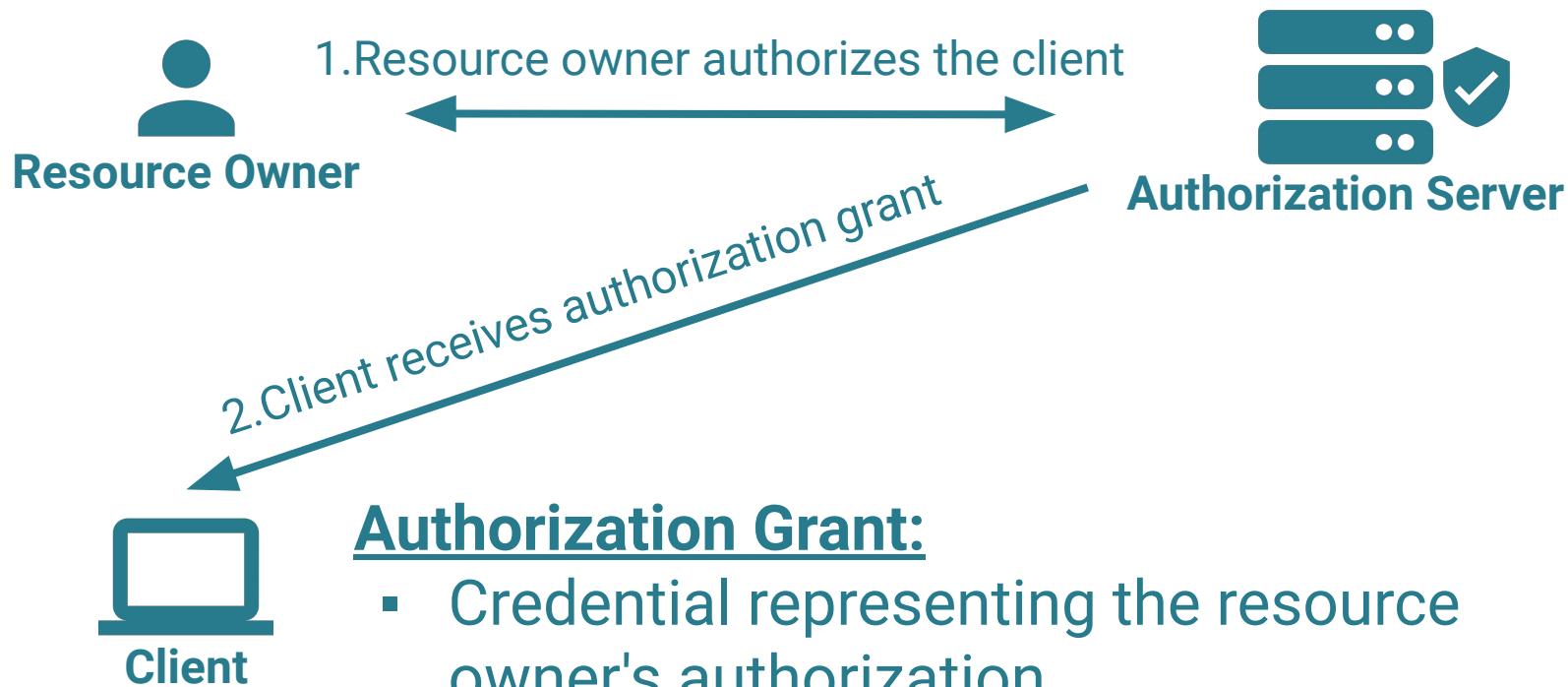
# Basic OAuth 2.0 Protocol Flow



# Protocol Flow (1): Resource owner authorizes client



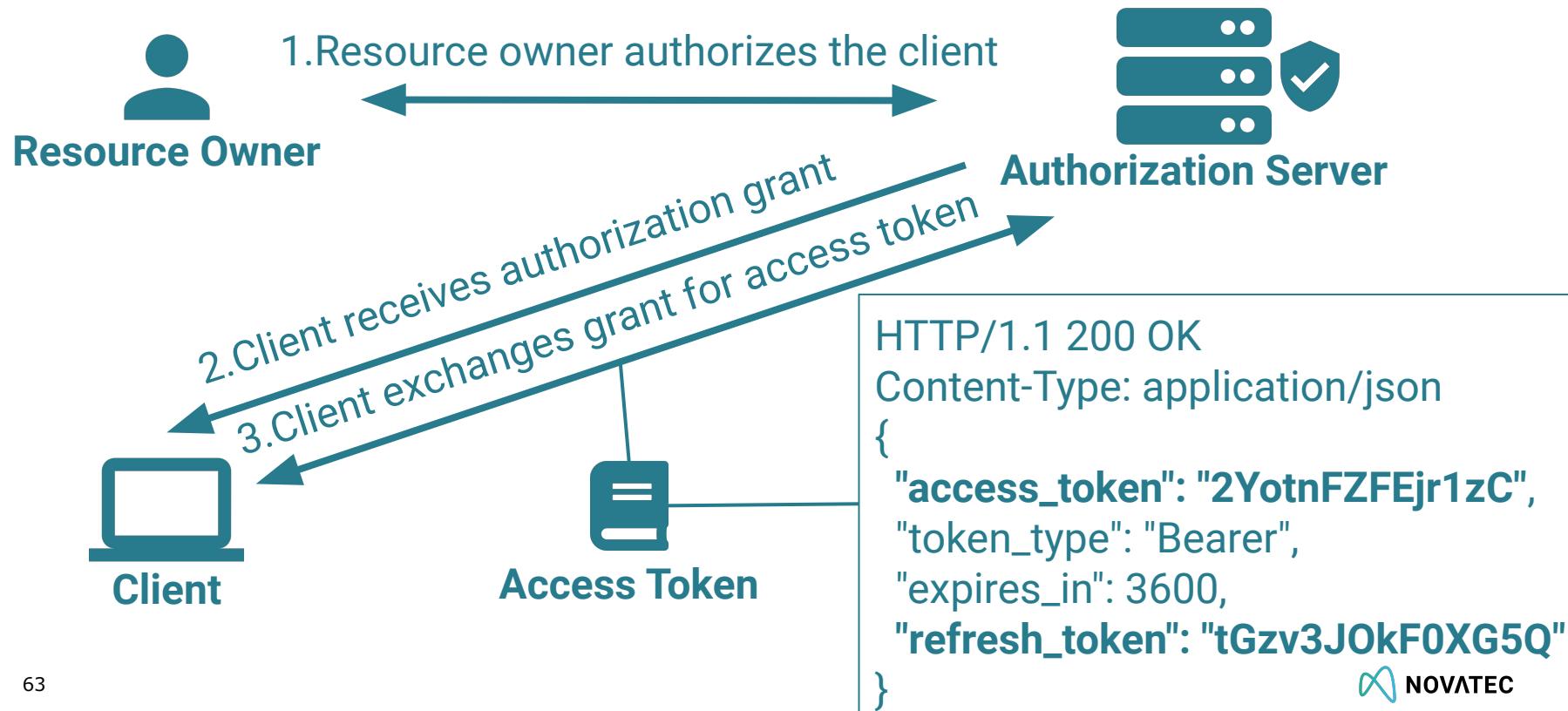
# Protocol Flow (2): Client receives authorization grant



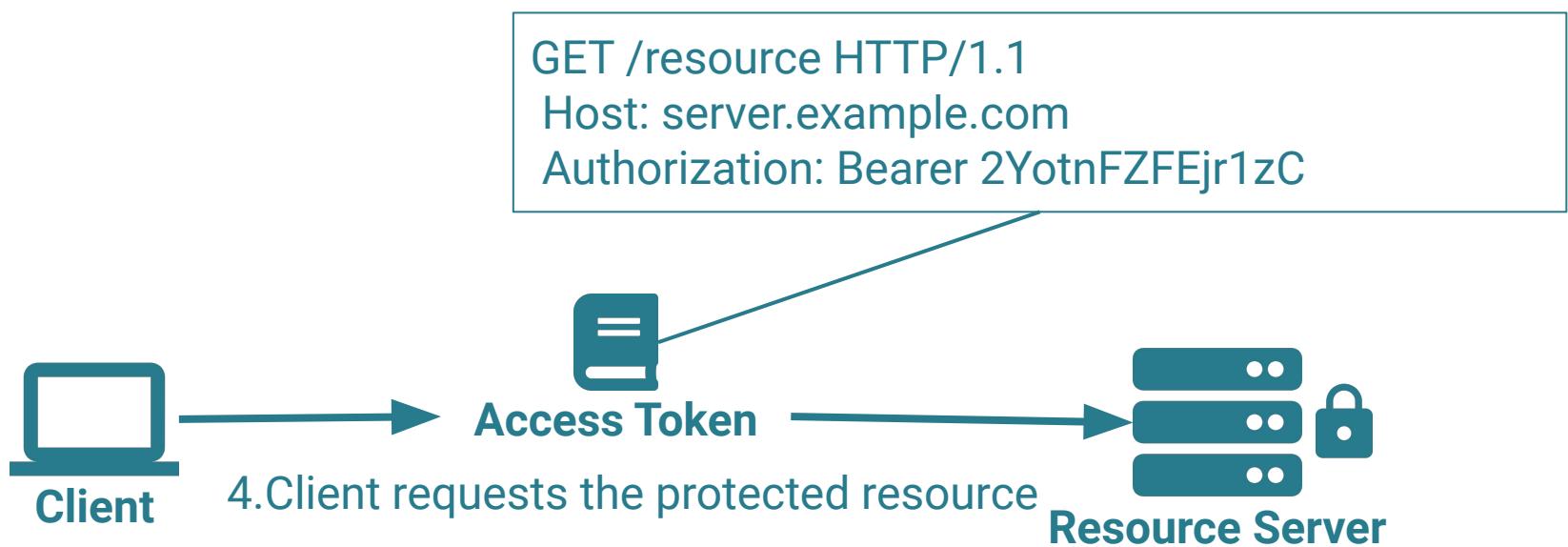
## Authorization Grant:

- Credential representing the resource owner's authorization
- Exchange for access token

# Protocol Flow (3): Client exchanges grant for access token



# Protocol Flow: Client requests the protected resource

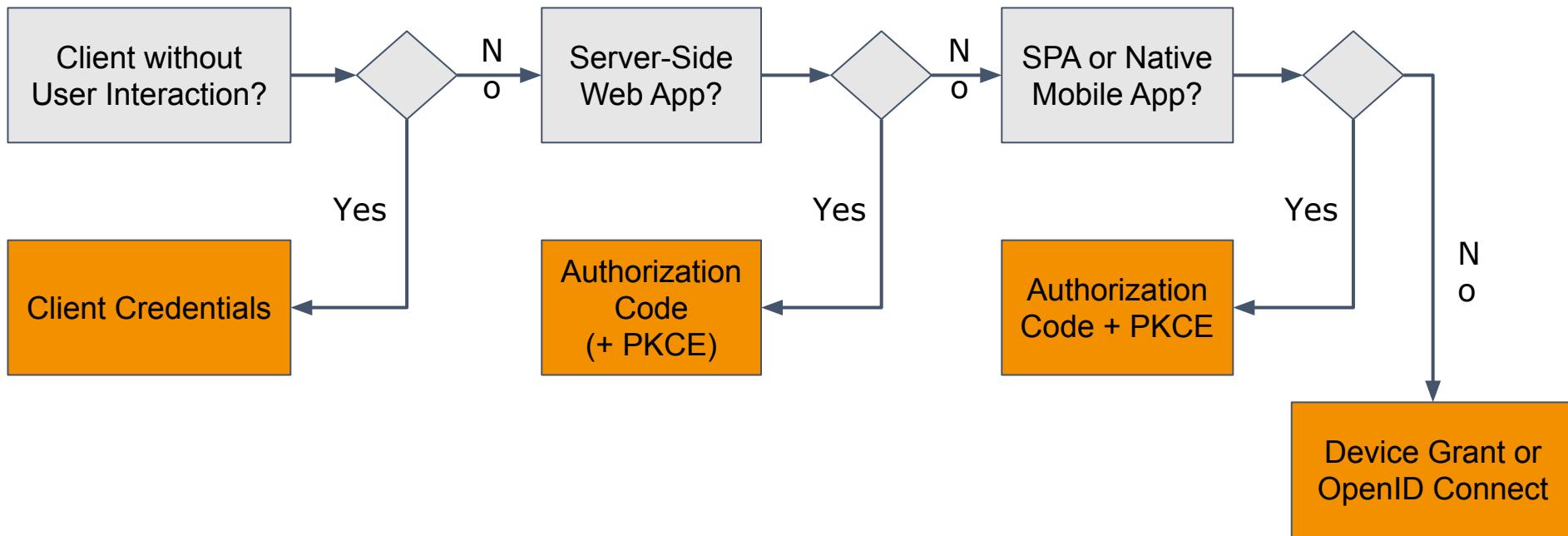


---

# Demo Time

## Authorization Code Grant Flow in Action

# Which Authorization Grant should I Use ?



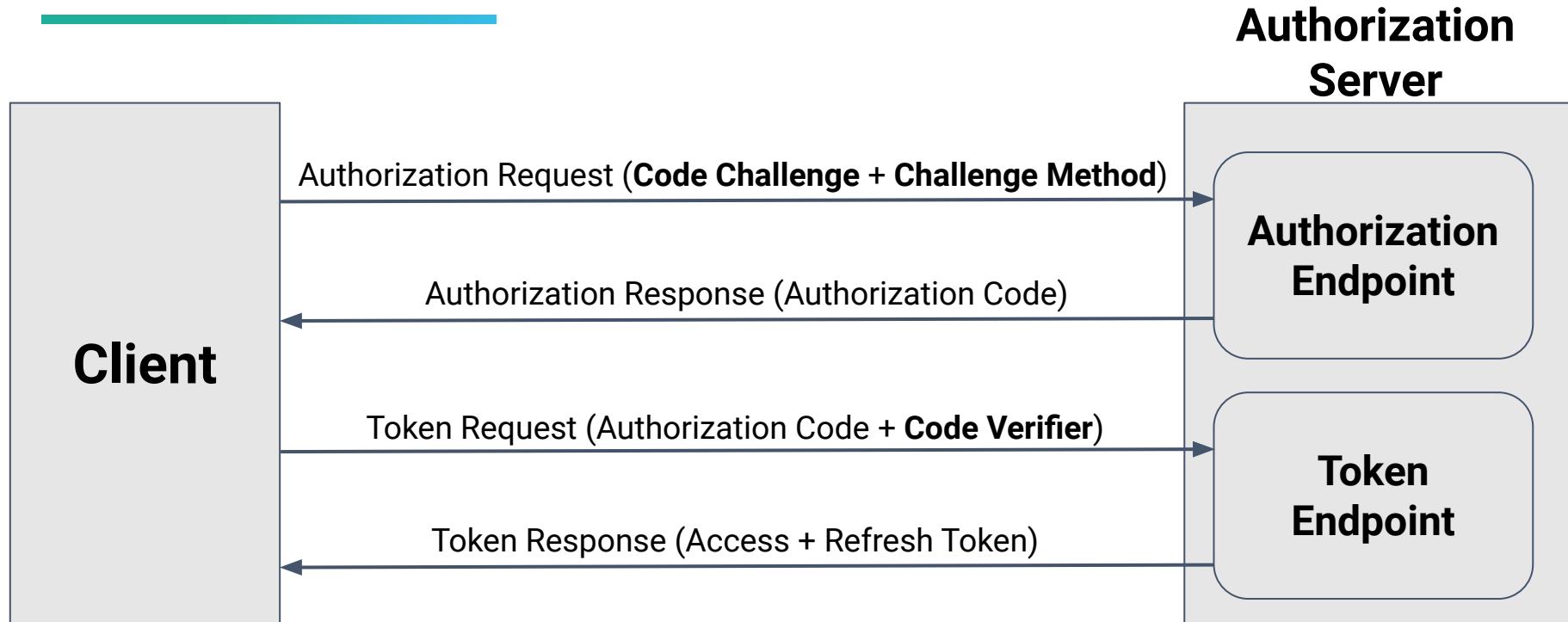
# OAuth 2.1 Authorization Grant Types

Resource Owner	Client Type	Authorization Grant	Refresh Token
✓	Web Client (Confidential)	Authorization Code + PKCE	✓
✓	Mobile Client (Public)	Authorization Code + PKCE	✓
✓	SPA Client (Public)	Authorization Code + PKCE	✓
✗	Resource Owner = Client (Confidential)	Client Credentials	✗

<https://datatracker.ietf.org/doc/draft-ietf-oauth-security-topics>

<https://datatracker.ietf.org/doc/draft-parecki-oauth-v2-1>

# Authorization Code + PKCE



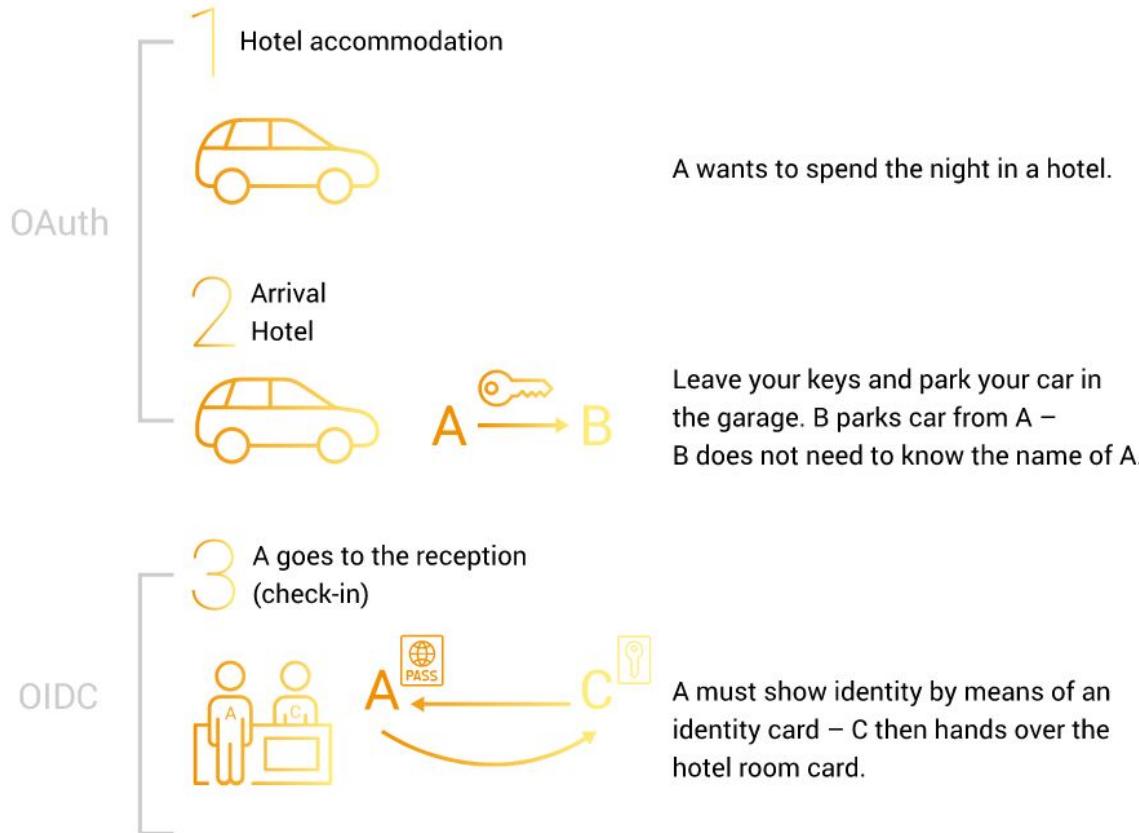


---

## OpenID Connect 1.0

[https://openid.net/specs/openid-connect-core-1\\_0.html](https://openid.net/specs/openid-connect-core-1_0.html)

# OAuth 2.0 is NOT an Authentication Protocol!



# OpenID Connect 1.0 Standards Layer

---

OpenID Connect 1.0

OAuth 2.0 Authorization Framework (RFC 6749)

Javascript Object Signing and Encryption (JOSE)

JSON Web Token (JWT)

JSON Web Signature (JWS)

JSON Web Encryption (JWE)

JSON Web Key (JWK)

JSON Web Algorithms (JWA)

# JSON Web Algorithms (JWA)

---

- Cryptographic algorithms and identifiers for JWS, JWE, and JWK specifications
- Digital Signatures and MACs
- Algorithms for Key Management
- Algorithms for Content Encryption
- Algorithms for Keys

<https://tools.ietf.org/html/rfc7518>

# JSON Web Key (JWK)

---

```
{"keys": [  
    {"kty":"EC",  
     "crv":"P-256",  
     "x":"MKBCTNIcKUSDii11ySs3526iDZ8AiTo7Tu6KPAqv7D4",  
     "y":"4Etl6SRW2YiLUrN5vfVHuhp7x8PxltmWWIbbM4IFyM",  
     "use":"enc",  
     "kid":"1"},  
    {"kty":"RSA",  
     "n": "0vx7agoebGcQSuuPiLJXZptN9nn...",  
     "e":"AQAB",  
     "alg":"RS256",  
     "kid":"2011-04-29"}]}
```

<https://tools.ietf.org/html/rfc7517>

# JSON Web Signature (JWS)

---

- JSON Web Signature (JWS) represents content secured with digital signatures or Message Authentication Codes (MACs) using JSON-based data structures
- A document using JWS can answer two questions about the JSON payload:
  - Has the JSON object been altered after creation?
  - Who created this JSON object?

<https://tools.ietf.org/html/rfc7515>

# JSON Web Encryption (JWE)

---

- Data structure representing an encrypted and integrity-protected message
- As of July 2019 only identity server of PingIdentity supports JWE
- NOT supported by Spring Security 5.x (See github issue 4435) !

<https://tools.ietf.org/html/rfc7516>

<https://github.com/spring-projects/spring-security/issues/4435>

# JSON Web Token (JWT)

---

- JSON Web Tokens consist of three parts separated by dots ("."), which are:
  - Header
  - Payload
  - Signature
- Each part is Base64Url encoded
- Signature supports symmetric or asymmetric algorithms (e.g. HMAC or RSA)
- Signature = HMACSHA256(  
base64UrlEncode(header) + "." + base64UrlEncode(payload), secret)

<https://tools.ietf.org/html/rfc7519>

<https://tools.ietf.org/html/draft-ietf-oauth-jwt-bcp>

<https://tools.ietf.org/html/draft-ietf-oauth-proof-of-possession>

# JSON Web Token (JWT) - Decoded Form

---

```
{  
  "alg": "RS256",  
  "typ": "JWT",  
  "kid": "Ip_FcMZ8D7U6EEUCiZyWAF21NcwjX_ddwJ5a3eCPMwQ"  
}  
  
{  
  "exp": 1571745342,  
  "iat": 1571745042,  
  "iss": "http://localhost:8080/auth/realms/workshop",  
  "aud": ["library-service","account"],  
  "sub": "08d3bcaa-5ffd-4b8d-909e-bb567881384b"  
}
```

HEADER

PAYLOAD

# <https://jwt.io>

ALGORITHM RS256

Encoded PASTE A TOKEN HERE

```
eyJhbGciOiJSUzI1NiIsInR5cCIgOiAiSl0Uiwi  
a2lkIiA6ICJscF9GY01a0EQ3VTZFRVVDaVp5V0FG  
MjFOY3dqWF9kZHdKNWEzZUNQTXdRIn0.eyJqdGki  
OiI1YjkxYTNmYS03MWVhLTQ2YzktyFkMS0xYzM0  
ZGYwNTVkJZWIiLCJleHAi0jE1NzE3NDUzNDIiSm5i  
ZiI6MCwiaWF0IjoxNTcxNzQ1MDQyLCJpc3Mi0iJo  
dHRwOi8vbG9jYWxob3N00jgwODAvYXV0aC9yZWFs  
bXMvd29ya3Nob3AiLCJhdWQi0lsibGlicmFyeS1z  
ZXJ2aNWN1iwiYWNjb3VudCJdLCJzdWIi0iIwOGQz  
YmNhYS01ZmZkLTRi0GQtOTA5ZS1iYjU2Nzg4MTM4  
NGIiLCJ0eXAiOiJCZWFyZXIiLCJhenAiOiJsaWJy  
YXJ5LWNsaWVuDCisImF1dGhfdG1tZSI6MCwic2Vz  
c21vb19zdGF0ZSI6IjdmZTBkMTM1LWJjMzktNDYz  
MC04Zm5LTEyODA0ZTEwYTUyOCIsImFjciI6IjEi  
LCJhbGxvd2VklW9yaWdpbnMiolsiaHR0cDovL2xv  
Y2FsaG9zdDo5MDkwIl0sInJlYWxtX2FjY2VzcyI6  
eyJyb2xlcyI6WyJsaWJyYXJ5X2FkbWluIiwb2Zm  
bGluZV9hY2Nlc3MiLCJ1bWFfYXV0aG9yaXphdGlv  
biJdfSwicmVzb3VbyY2VfYWnjZXNzIjp7ImFjY291  
bnQiOnsicm9sZXMi0lsibWFuYWd1LWFjY291bnQi  
LcI+YW5hZ2UitYWMib3VudC1ceW5rcyTcTn7n7Yct
```

Decoded EDIT THE PAYLOAD AND SECRET

HEADER: ALGORITHM & TOKEN TYPE

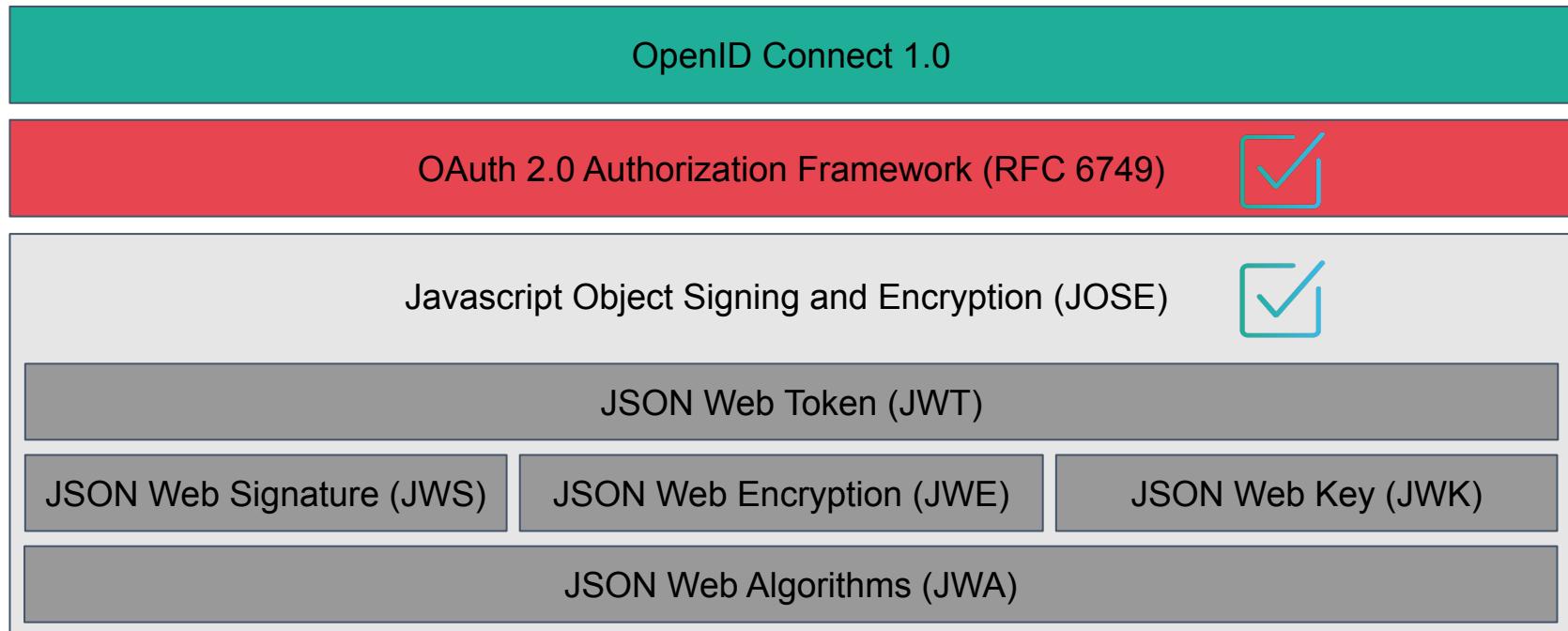
```
{"alg": "RS256",  
 "typ": "JWT",  
 "kid": "1p_FcMZ8D7U6EEUCiZyWAF21NcwjX_ddwJ5a3eCPMwQ"  
}
```

PAYLOAD: DATA

```
{  
  "jti": "5b91a3fa-71ea-46c9-b1d1-1c34df055deb",  
  "exp": 1571745342,  
  "nbf": 0,  
  "iat": 1571745042,  
  "iss": "http://localhost:8080/auth/realms/workshop",  
  "aud": [  
    "library-service",  
    "account"  
  ],  
  "sub": "08d3bcaa-5ffd-4b8d-909e-bb567881384b",  
  "typ": "Bearer",  
  "azp": "library-client",  
  "auth_time": 0,  
  "session_state": "7fe0d135-bc39-4630-8fc9-12804e10a528",  
  "acr": "",  
  "allowed-origins": [  
    "http://localhost:9090"  
  ]  
}
```

# OpenID Connect 1.0 Standards Layer

---



# OpenID Connect 1.0 (OIDC)

---

- Based on OAuth 2.0
- Additions:
  - ID Token (JWT format is mandatory)
  - User Info Endpoint (Mandatory)
  - Hybrid Grant Flow (Mandatory)
  - OpenID Provider Configuration Information (Discovery, Optional)

[https://openid.net/specs/openid-connect-core-1\\_0.html](https://openid.net/specs/openid-connect-core-1_0.html)

[https://openid.net/specs/openid-connect-registration-1\\_0.html](https://openid.net/specs/openid-connect-registration-1_0.html)

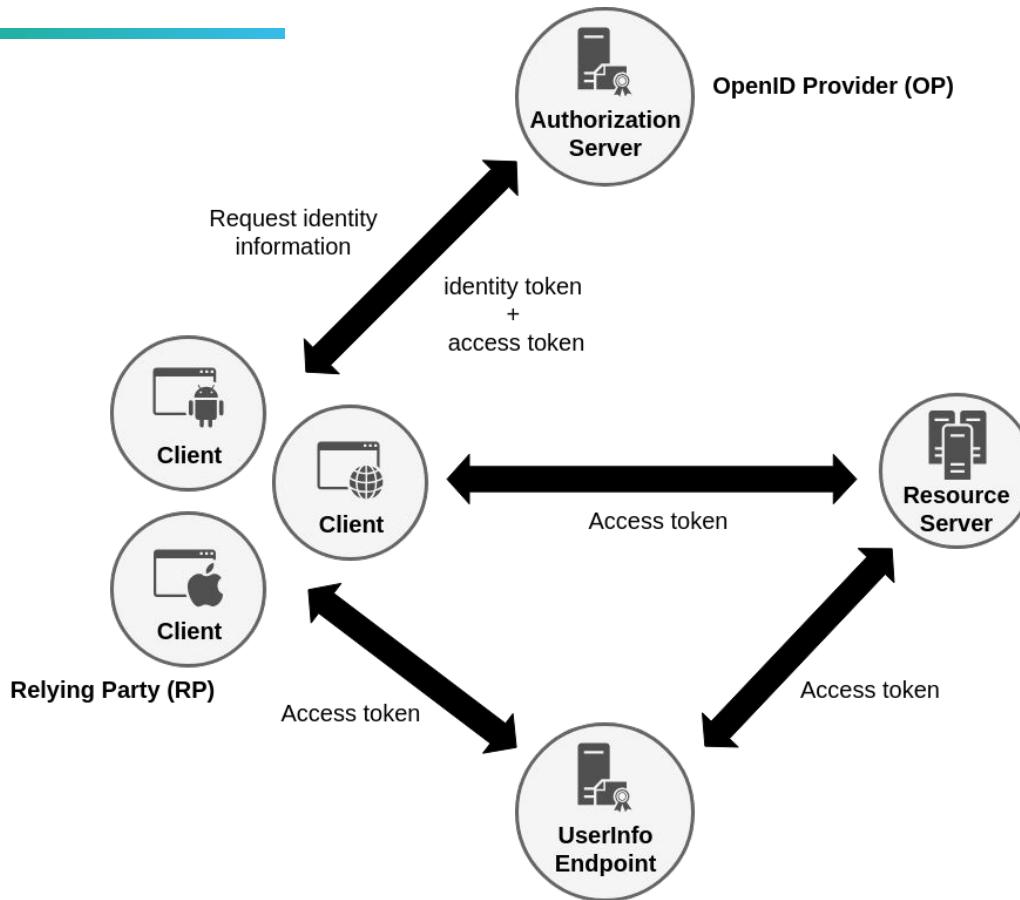
[https://openid.net/specs/openid-connect-discovery-1\\_0.html](https://openid.net/specs/openid-connect-discovery-1_0.html)

# OpenID Connect 1.0 Claims

---

Claim	Required	Description
iss	<input checked="" type="checkbox"/>	Issuer Identifier
sub	<input checked="" type="checkbox"/>	Unique Subject Identifier
aud	<input checked="" type="checkbox"/>	Target audience(s) of an ID Token
exp	<input checked="" type="checkbox"/>	Expiration time
iat	<input checked="" type="checkbox"/>	Time at which the JWT was issued
auth_time	<input checked="" type="checkbox"/>	Time of End-User authentication
nonce		Used to associate a client with an ID Token

# OpenID Connect 1.0 Roles



# OpenID Connect 1.0 Discovery

```
{  
  "issuer": "https://access-me.eu.auth0.com/",  
  "authorization_endpoint": "https://access-me.eu.auth0.com/authorize",  
  "token_endpoint": "https://access-me.eu.auth0.com/oauth/token",  
  "userinfo_endpoint": "https://access-me.eu.auth0.com/userinfo",  
  "jwks_uri": "https://access-me.eu.auth0.com/.well-known/jwks.json",  
  "scopes_supported": [  
    "openid",  
    "profile"  
  ],...  
}
```

<https://access-me.eu.auth0.com/.well-known/openid-configuration>

# OpenID Connect 1.0 User Info Endpoint

GET /userinfo HTTP/1.1

Host: access-me.eu.auth0.com

Authorization: Bearer eyJhbGciOiJSUzI1N...

HTTP/1.1 200 OK

Content-Type: application/json; charset=utf-8

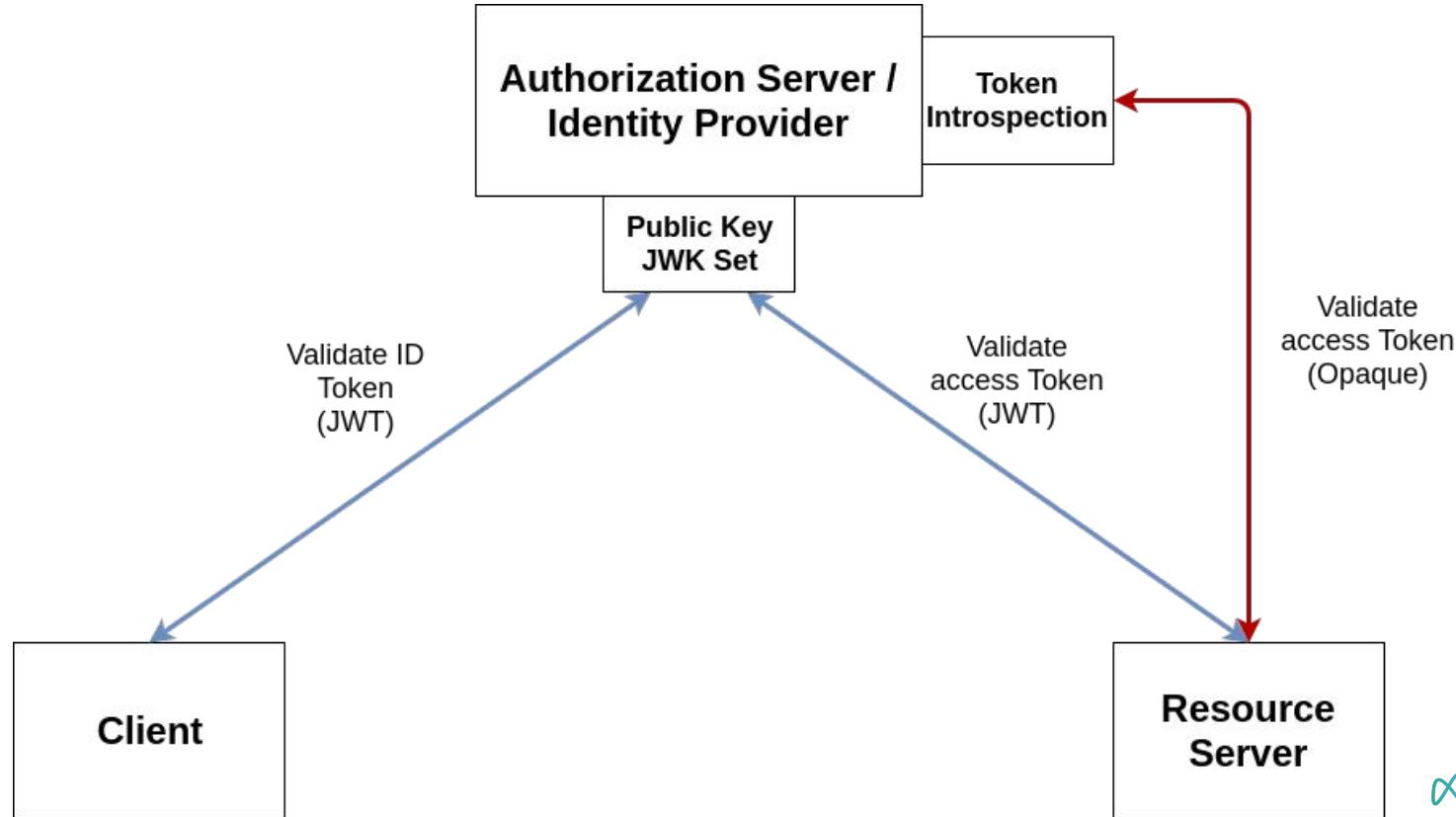
```
{  
  "email": "user@example.com",  
  "email_verified": true,  
  "sub": "auth0|5bc44fceb144eb0173391741"  
}
```

# OpenID Connect 1.0: Access Token Types

---

JWT Token (Self-contained)	Opaque Token (Reference)
<input checked="" type="checkbox"/> Offline-Validation (Signature/Expiration)	! Validation call to introspection-endpoint
<input checked="" type="checkbox"/> Contains all required information	! Additional call to get required information
<input checked="" type="checkbox"/> Protocol agnostic	! Bound to Http
! Cannot be revoked	<input checked="" type="checkbox"/> May be revoked
! Mandatory for Id Tokens	! Must not be used for Id Tokens
May be used for Access Tokens	May be used for Access Tokens

# Token Validation



# OpenID Connect Identity Providers

---

- RedHat/JBoss Keycloak (<https://www.keycloak.org>)
- Auth0 (<https://auth0.com>)
- Okta (<https://www.okta.com>)
- ForgeRock (<https://www.forgerock.com/platform/identity-management>)
- CloudFoundry UAA (<https://github.com/cloudfoundry/uaa>)
- PingFederate  
(<https://www.pingidentity.com/en/platform/single-sign-on/sso-overview.html>)
- Azure Active Directory  
(<https://azure.microsoft.com/en-us/services/active-directory>)
- ...

See: <https://openid.net/developers/certified/#OPServices>

---

# **OpenID Connect on the Server side (Resource Server)**

# Authentication in a single Resource server

---



GET / HTTP/1.1

Host: localhost:8080

Authorization: Bearer

eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1N...

---

# OpenID Connect on the Client side

# OAuth 2.1 Authorization Grant Types

Resource Owner	Client Type	Authorization Grant	Refresh Token
✓	Web Client (Confidential)	Authorization Code + PKCE	✓
✓	Mobile Client (Public)	Authorization Code + PKCE	✓
✓	SPA Client (Public)	Authorization Code + PKCE	✓
✗	Resource Owner = Client (Confidential)	Client Credentials	✗

<https://datatracker.ietf.org/doc/draft-ietf-oauth-security-topics>

<https://datatracker.ietf.org/doc/draft-parecki-oauth-v2-1>

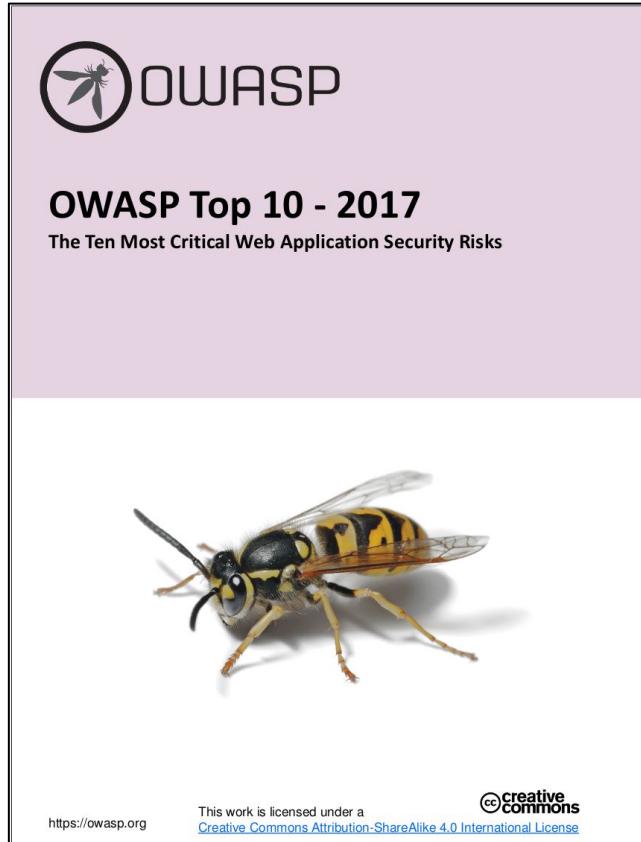
# OpenID Connect Libraries

---

- oidc-client (Javascript) <https://github.com/IdentityModel/oidc-client-js>
- angular-oauth2-oidc (TypeScript)  
<https://github.com/manfredsteyer/angular-oauth2-oidc>
- angular-auth-oidc-client (TypeScript)  
<https://github.com/damienbod/angular-auth-oidc-client>
- IdentityModel.OidcClient (C#/.Net)  
<https://github.com/IdentityModel/IdentityModel.OidcClient>
- Nimbus OAuth 2.0 SDK (Java)  
<https://connect2id.com/products/nimbus-oauth-openid-connect-sdk>
- OIDC RP library (Python) <https://github.com/openid/JWTConnect-Python-OidcRP>
- ...

See: <https://openid.net/developers/certified/#OPServices>

# A5: Broken Access Control



# Authorization - What can I access?

---



HTTP 403 - FORBIDDEN

# Authorization (Static Roles)

---

```
public class UserBoundaryService {  
  
    @PreAuthorize("hasRole('ADMIN')")  
    public List<User> findAllUsers() { . . . }  
}
```

# Authorization (Dynamic Permissions)

---

```
public class TaskBoundaryService {  
  
    @PreAuthorize("hasPermission(#taskId, 'TASK', 'WRITE')")  
    public Task findTask(UUID taskId) {...}  
}
```

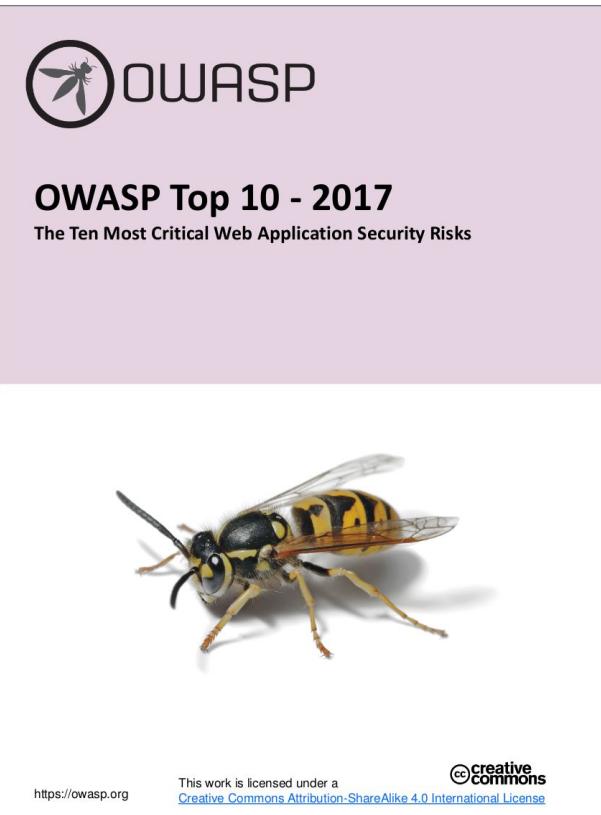
# Automated Authorization Testing

---

```
public class AuthorizationIntegrationTest {  
  
    @WithMockUser(roles = "ADMIN")  
    @Test  
    public void verifyfindAllUsersAuthorized() { ... }  
  
    @WithMockUser(roles = "USER")  
    @Test(expected = AccessDeniedException.class)  
    public void verifyfindAllUsersUnauthorized() { ... }  
}
```

# A1: Injection

---



# SQL Injection

---

```
SELECT balance FROM user WHERE username =  
    + " " + request.getParameter('userName')  
    + " ";
```

→ Parameter *userName*: Tom

```
SELECT balance FROM user WHERE username = 'Tom';
```

→ Parameter *userName*: Tom' or 1=1; --

```
SELECT balance FROM user WHERE username = 'Tom' or 1=1
```

# SQL Injection Defensive - Use Input Validation

---

```
@Entity
public class Person extends AbstractPersistable<Long> {

    @NotNull
    @Pattern(regexp = "^[A-Za-z0-9- ]{1,30}$")
    private String lastName;

    @NotNull
    @Enumerated(EnumType.STRING)
    private GenderEnum gender;

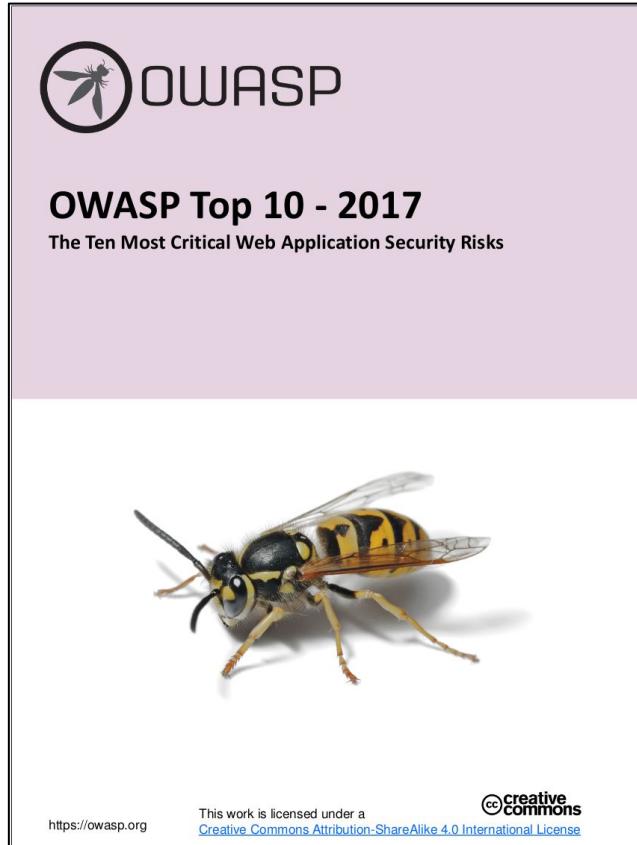
    ...
}
```

# SQL Injection Defensive - Prepared Statements

---

```
@Query(  
    "select u from User u where u.username = "  
    + " :username and u.password = :password")  
User findByUsernameAndPassword(  
    @Param("username") String username,  
    @Param("password") String password);
```

# A9: Using Components With Known Vulnerabilities



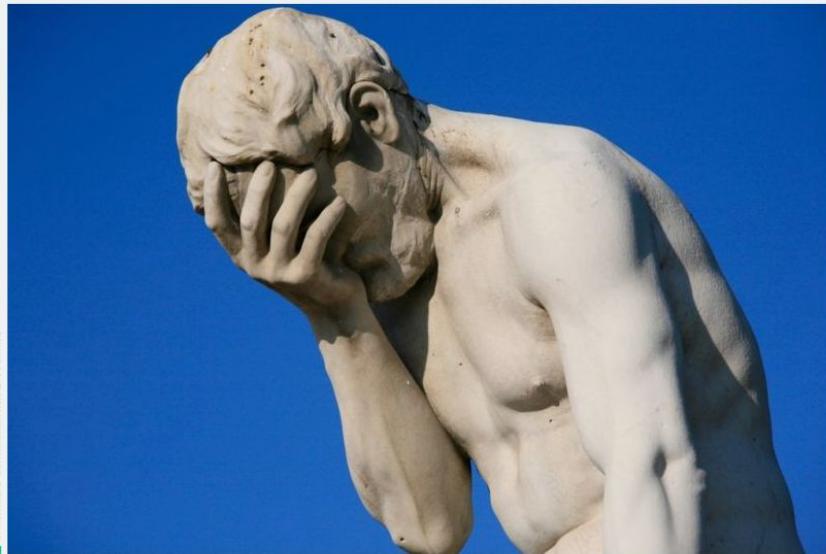
# Equifax Struts Vulnerability

BIZ & IT —

## Failure to patch two-month-old bug led to massive Equifax breach

Critical Apache Struts bug was fixed in March. In May, it bit ~143 million US consumers.

DAN GOODIN - 9/14/2017, 5:12 AM



[Enlarge](#)

The Equifax breach that exposed sensitive data for as many as 143 million US consumers was accomplished by exploiting a Web application vulnerability that had been patched more than two months earlier, officials with the credit reporting service said Thursday.

# OWASP Dependency Check

---

- Detects Vulnerabilities in Project Dependencies
- Supports Java and .NET applications
- Experimental: Python, Ruby, PHP, Node.js
- Command line, Ant, Maven, Gradle, Jenkins, SBT

<https://github.com/jeremylong/DependencyCheck>

---

# What about the Cloud?

## **“Good old friends” + more**

---

- CSRF, XSS, SQL Injection, ...
- Distributed DoS
- Economic DoS

# Weak Passwords

Microsoft Azure

All services

**Overview** **Featured**

**Categories**

All	 Virtual machines	 App Services	 Storage accounts	 SQL databases	 Azure Database for	 Azure Cosmos DB	 Kubernetes services	 Function App
General								
Compute								
Networking	 Virtual networks	 Azure Active Directory	 Resource groups	 Monitor	 Advisor	 Security Center	 Cost Management	 → All services
Storage								
Web								
Mobile								
Containers								
Databases								
Analytics								
Blockchain								
AI + machine learning								
Internet of things								
Mixed reality								
Integration								
Identity								

**Free training from Microsoft** [See all](#)



**Core Cloud Services - Azure architecture and service guarantees**  
9 units • 45 min  
Azure provides a global network of secure datacenters you can deploy your services into. Learn about the physical architecture of Azure, how redundancy is provided, and what sort of service guarantees

Microsoft modules

[Start](#)



**Core Cloud Services - Manage services with the Azure portal**  
9 units • 1 hr 13 min  
Tour the Azure portal features and services, and customize the portal.

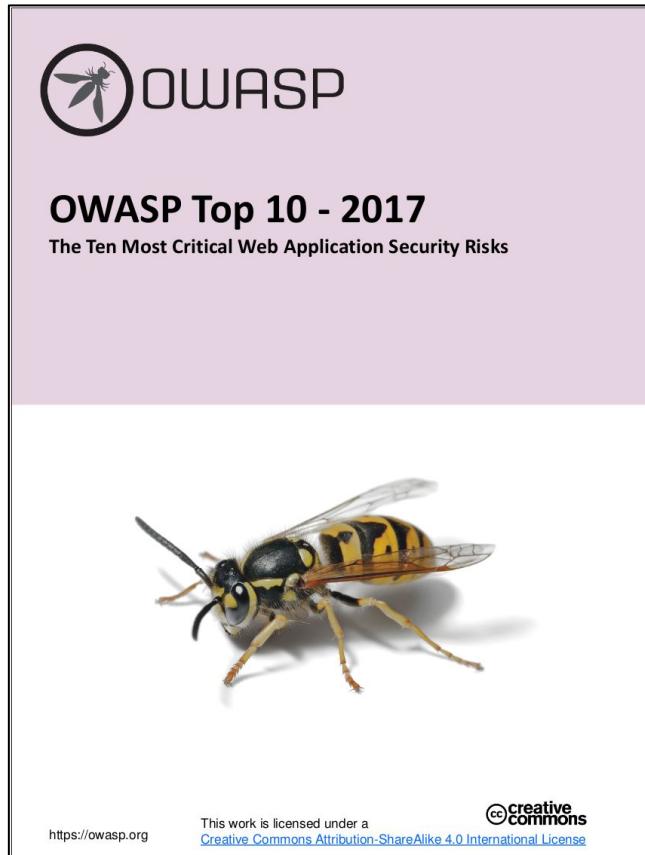
[Start](#)



**Cloud Concepts - Principles of cloud computing**  
10 units • 1 hr 2 min  
Explore the core concepts of cloud computing and how it can help your business.

[Start](#)

# A3: Sensitive Data Exposure



# EU General Data Protection Regulation (GDPR)

---

“...should adopt internal policies and implement measures which meet in particular the principles of **data protection by design** and **data protection by default**”

<http://eur-lex.europa.eu/legal-content/EN/TXT/HTML/?uri=CELEX:32016R0679>

# Key Management

---



# Key Management

---

- Azure Key Vault
- AWS Secrets Manager
- Google Cloud HSM
- CloudFoundry CredHub
- Hashicorp Vault



# Rotate, Repair, Repave

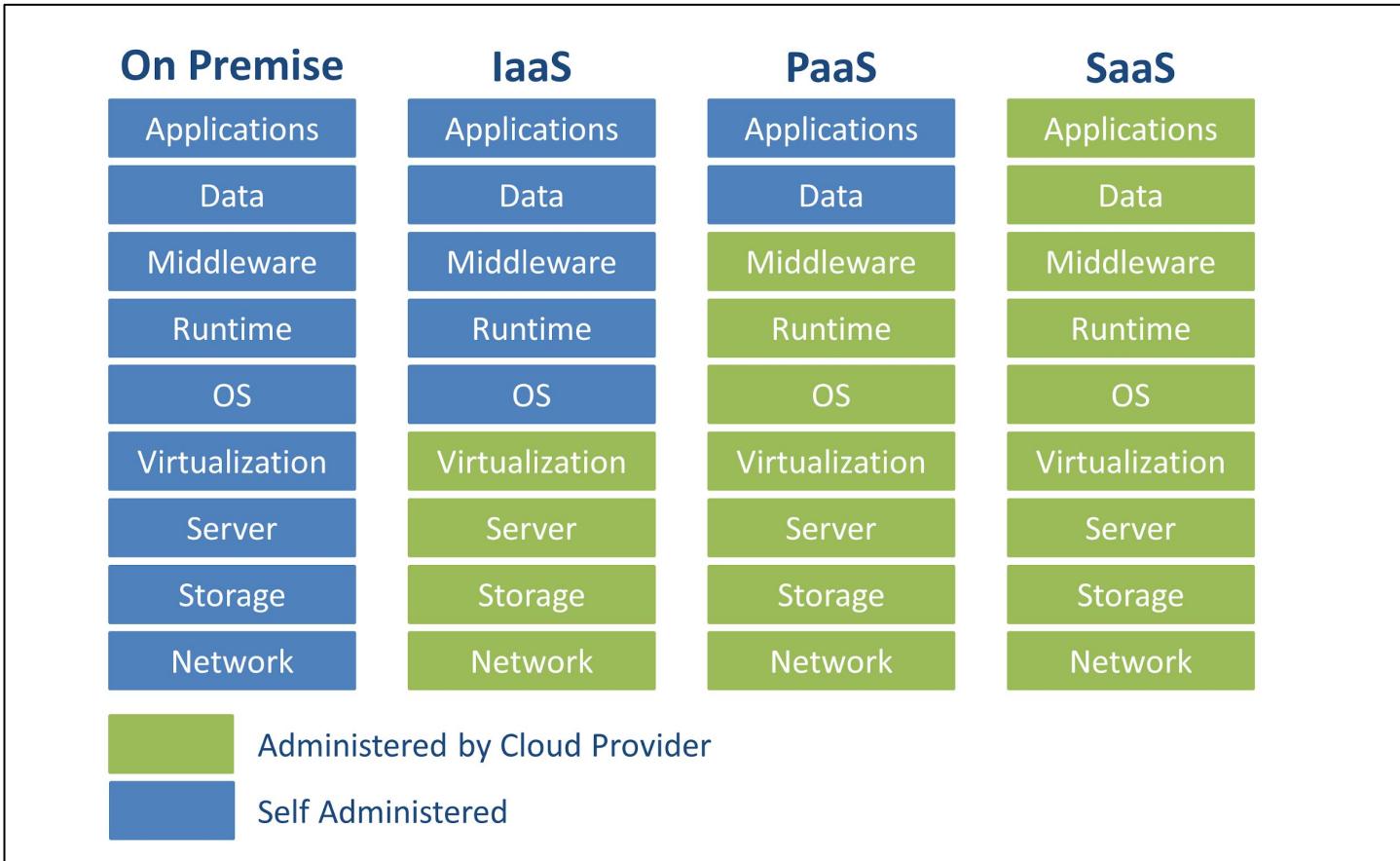
---

“What if every server inside my data center had a maximum lifetime of two hours? This approach would frustrate malware writers...”

**Justin Smith (Chief Security Officer at Pivotal)**

<https://thenewstack.io/cloud-foundrys-approach-security-rotate-repair-repave>

# Cloud Service Models



---

# Spring Security 5 Basics

# Spring Security 5

---

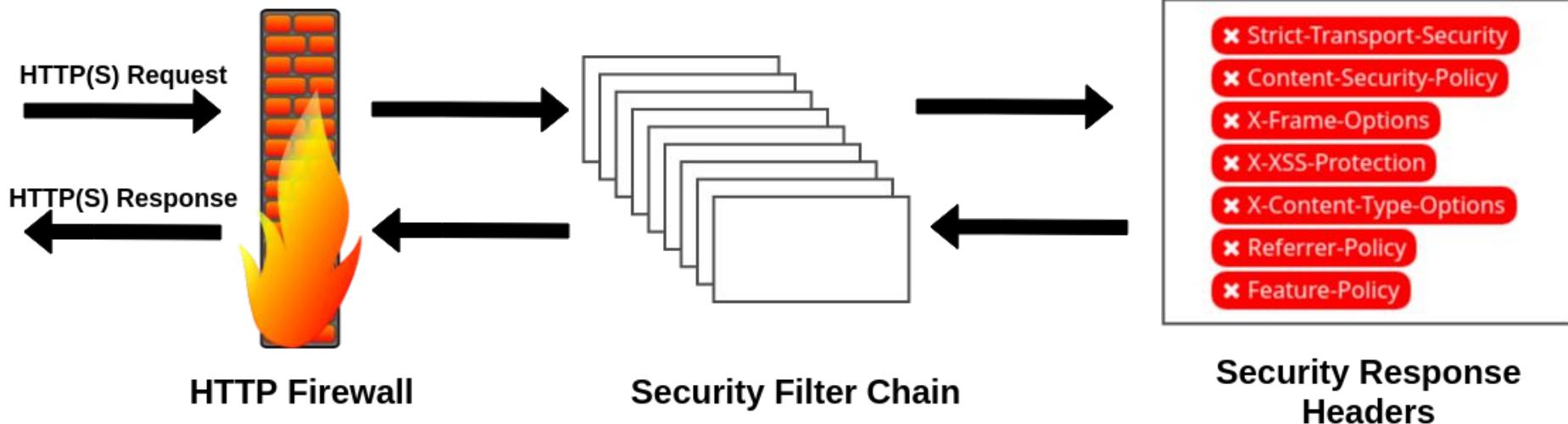
- Authentication & Authorization
- Password Encoding
- Support for Servlet & Reactive Web Applications
- Support for OAuth 2.0
- Support for OpenID Connect 1.0, JWT and JOSE (JWS/JWE/JWK)
- Testing Support for Auth/Authz/JWT

# Spring Security 5 - Secure by Default

---

- Authentication required for all HTTP endpoints
- Session Fixation Protection
- Session Cookie (HttpOnly, Secure)
- CSRF Protection
- Security Response Header

# Spring Security High Level View



# New Spring Security 5.x OIDC Technology Stack

---

**spring-boot-starter-oauth2-client**

**spring-boot-starter-oauth2-resource-server**

**spring-security-oauth2-jose**



**com.nimbusds:oauth2-oidc-sdk**

**spring-boot**

<https://connect2id.com/products/nimbus-oauth-openid-connect-sdk>

---

# Practice

<https://andifalk.gitbook.io/cloud-security-workshop>

<https://github.com/andifalk/cloud-security-workshop>



**Andreas Falk**

Managing Consultant

Mobil: +49 151 46146778

E-Mail: [andreas.falk@novatec-gmbh.de](mailto:andreas.falk@novatec-gmbh.de)

## **Novatec Consulting GmbH**

Dieselstraße 18/1

D-70771 Leinfelden-Echterdingen

T. +49 711 22040-700

[info@novatec-gmbh.de](mailto:info@novatec-gmbh.de)

[www.novatec-gmbh.de](http://www.novatec-gmbh.de)