



Pontificia Universidad Católica de Chile
Escuela de Ingeniería
Departamento de Ciencia de la Computación

Sesión

IIC2513 - Tecnologías y Aplicaciones Web

Sebastián Vicencio R.
2do Semestre 2020

HTTP stateless

Un request independiente
del siguiente

HTTP stateless

¿Cómo entonces podemos **iniciar sesión** y **ser reconocidos** en **visitas posteriores?**

Sabemos que ocurre...

HTTP stateless



HTTP cookies

Datos que servidor envía a
browser

Cookies

¿Para qué sirven?

Proveer una manera en que servidor sepa
si **dos requests vienen del mismo browser**

Están asociadas a un dominio

Cookies

¿Cómo funcionan?

Formato: **key=value**

1. **Servidor envía cookies** en response y browser las almacena
2. Cookies son incluidas por browser en **siguientes requests**

Cookies

Response headers

Servidor envía **response que incluye headers**

```
Set-Cookie: first_key=first_value
```

```
Set-Cookie: second_key=second_value
```


Cookies

Request headers

Browser almacena cookies

Las incluye en un header en **requests subsecuentes**

```
Cookie: first_key=first_value; second_key=second_value
```

Cookies



Cookies

Atributos

Response puede **incluir** una serie de **atributos**

1. Expires
2. Secure
3. HttpOnly
4. Path

```
Set-Cookie: id=a3fWa; Expires=Wed, 21 Oct 2021 07:28:00 GMT; Secure; HttpOnly
```

Sesión

Un uso de cookies

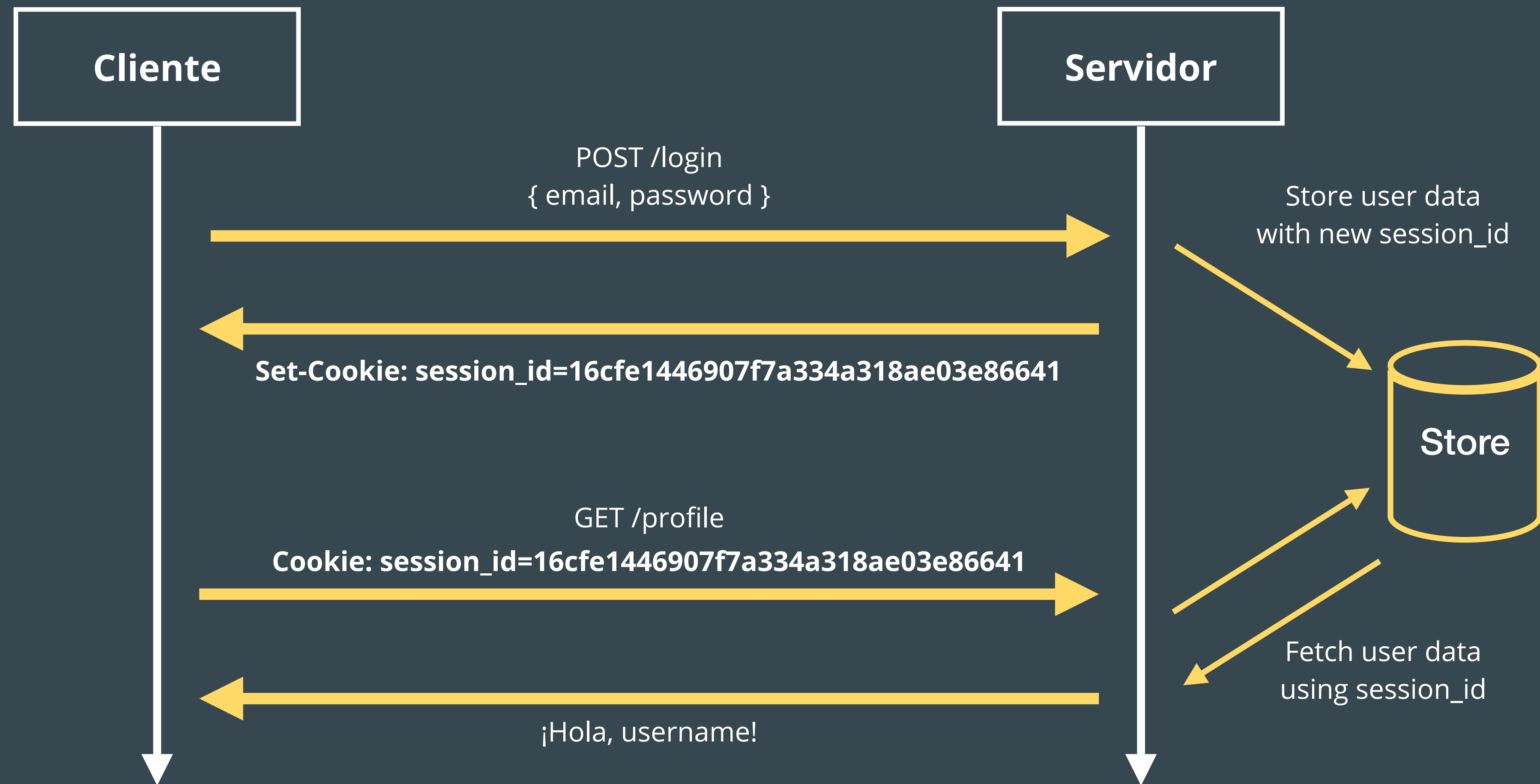
Sesión

¿Qué es?

Forma de **reconocer** específicamente un **usuario en requests sucesivos**

1. Datos de sesión en una o más cookies
2. Cookie sólo incluye un id de sesión

Sesión



¿Y en koa?

koa-session

koa-session

¿Cómo lo usamos?

```
ctx.session;
```

```
ctx.session.firstKey = 'firstValue';  
console.log(ctx.session.firstKey);  
// 'firstValue'
```

Leer y escribir propiedades según necesidad

Referencias

- [MDN - Using HTTP cookies](#)
- [RFC6265 - HTTP State Management Mechanism](#)
- [How HTTP Cookies Work](#)