



PONTIFICIA UNIVERSIDAD CATÓLICA DE CHILE  
ESCUELA DE INGENIERÍA  
DEPARTAMENTO DE CIENCIA DE LA COMPUTACIÓN

# HTTP

STATE IN A STATELESS PROTOCOL

RAÚL MONTES T.

# HTTP es *stateless*

---

- Interacción request → response
- Cada request–response es completamente independiente de los anteriores
- No importa si es el mismo cliente, o se usa la misma conexión TCP

# Entonces cómo... ?

---

→ POST /login

← ¡Inicio de sesión exitoso!

→ GET /index

← ¡Bienvenido, Usuario!

# HTTP Cookies

---



# HTTP Cookies

---

- Son datos en la forma *key* → *value*
- Server puede, en la *respuesta* HTTP, *solicitar el almacenamiento en el cliente* de estas piezas de información
- *Cliente* se obliga a *enviar* estos datos en *cada request futuro* al servidor
- Tanto la solicitud de almacenar *cookies* como el envío de las mismas se realiza mediante *headers* HTTP

# HTTP Cookies

---

→ POST /login

← ¡Inicio de sesión exitoso!

Set-Cookie: some\_info=123

→ GET /index

Cookie: some\_info=123

← ¡Bienvenido, Usuario!

# HTTP Cookies

---

- Al enviar una **cookie** en la respuesta se puede **configurar**:
  - restringiéndola a un **dominio** específico  
`Domain=.wican.org`
  - restringiéndola a un **path** específico  
`Path=/ongs`
  - con **fecha de expiración**, luego de la cual el cliente puede removerla  
`Expires=Tue, 12 Sep 2016 14:20:00 GMT`
  - con restricciones adicionales  
`HttpOnly; Secure`
- El cliente sólo enviará la cookie en los requests que cumplan con las restricciones

# HTTP Cookies

---

## Ejemplo

header en response...

```
Set-Cookie: some_info=123; Expires=Tue, 12 Sep 2016  
14:20:00 GMT; Path=/; Domain=.wican.org; HttpOnly; Secure
```

header en requests sucesivos...

```
Cookie: some_info=123; some_other_info=blah
```



# HTTP Cookies

---

Estándar: RFC6265

# La sesión

---

Plataformas usualmente proveen un manejo de “**sesión**” de más alto nivel, construido en base a las **cookies**

La mayoría ofrecen una o más de las siguientes alternativas:

- Se almacenan **todos los datos** de sesión en una o más **cookies**
- Se utiliza **sólo una cookie** para almacenar un **identificador** de la sesión; los **datos** se almacenan en **servidor** (memoria, BD, etc.)

# En Rails

---

Default es almacenar **todos los datos** de sesión en una **cookie**, pero **encriptada**

También existen las opciones de **generar** un **session\_id** almacenado en una cookie, y persistir el **hash de sesión** en **ActiveRecord** o **memcached** o...  
“custom”

# En Rails

---

Todos los controladores tendrán disponible `session` como un método que retorna un hash read/write

```
session
@attribute = session[key]
session[key] = some_value
```