

# IIC3253

Seguridad en la Web



fintual.cl



Hola de nuevo 

Email

ejemplo@ejemplo.com

Contraseña

.....



Entrar



fintual.cl



Hola Martín 

Invirtiendo fácil hace 477 días

+ Nuevo Objetivo

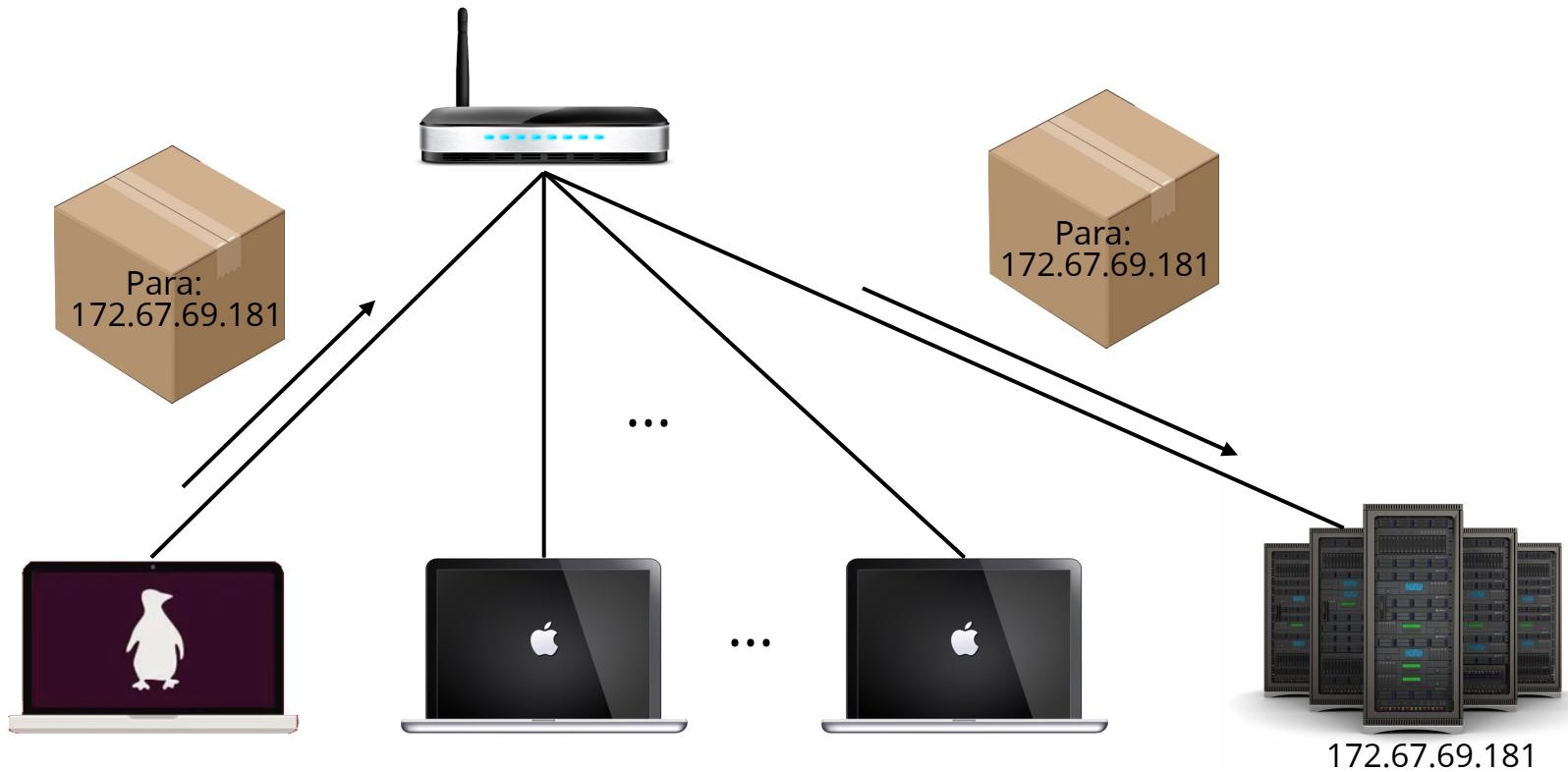
 Invertir más



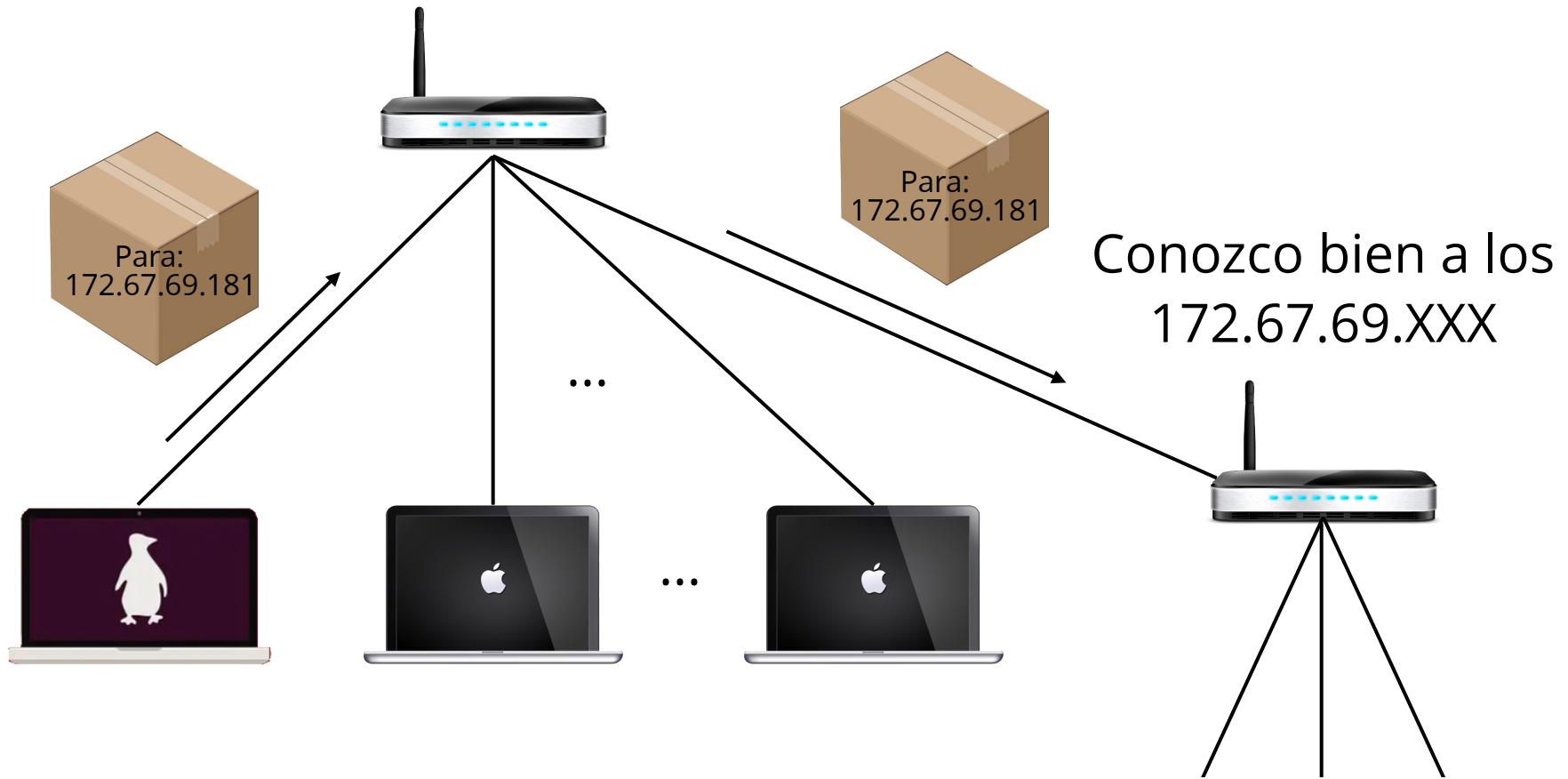
**THAT'S IT?**

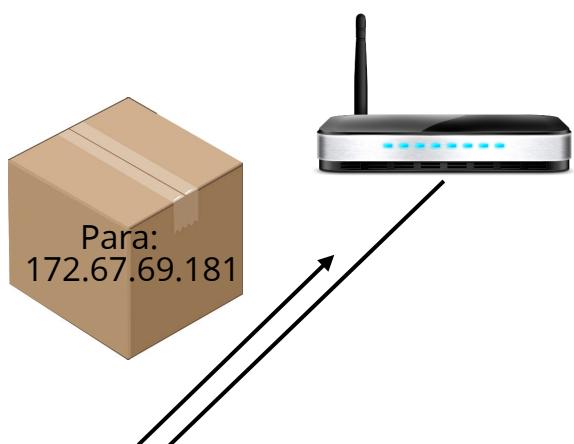
**TOO EASY.**

## Conozco esa dirección?



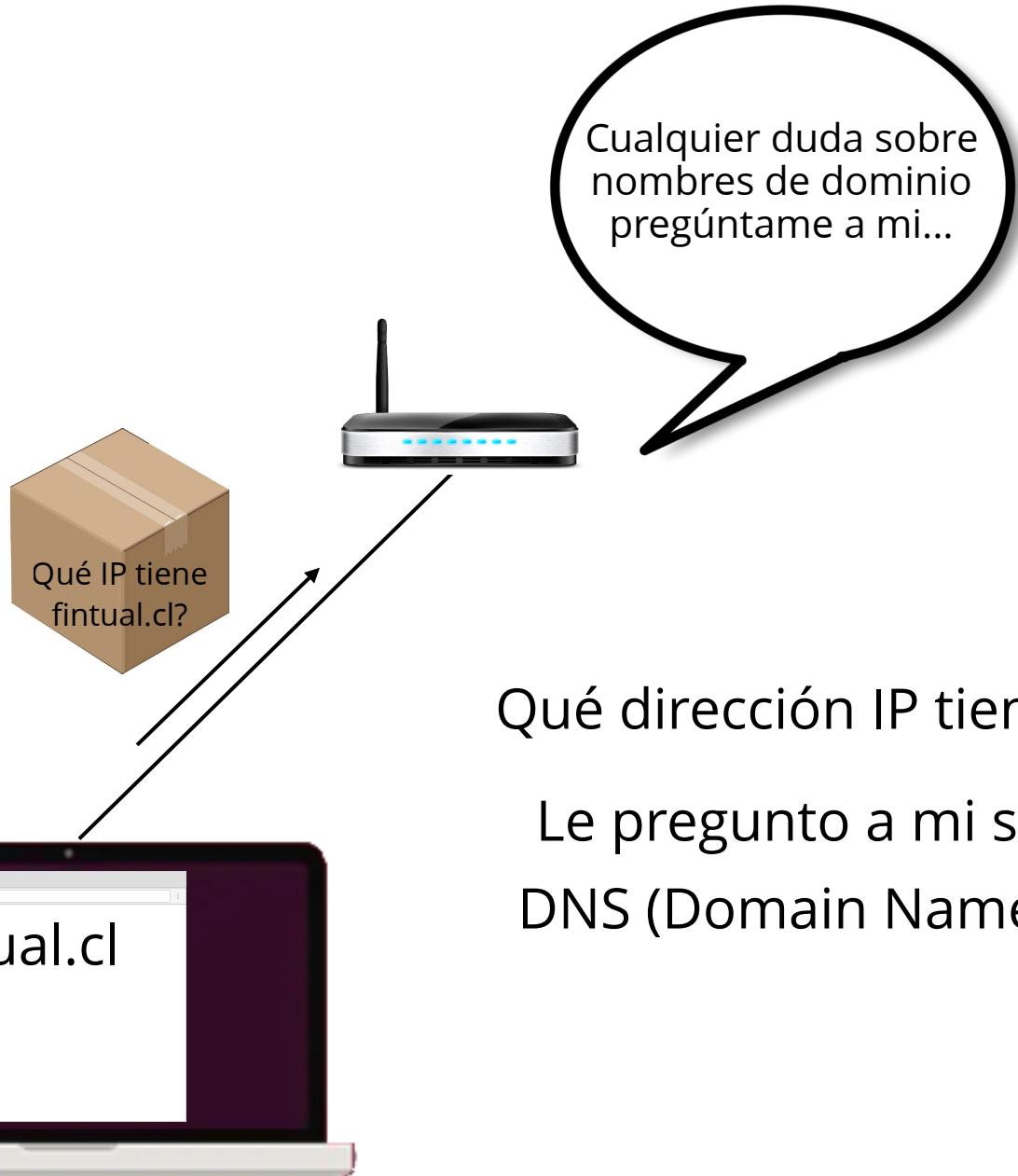
Conozco a alguien que  
conozca esa dirección?

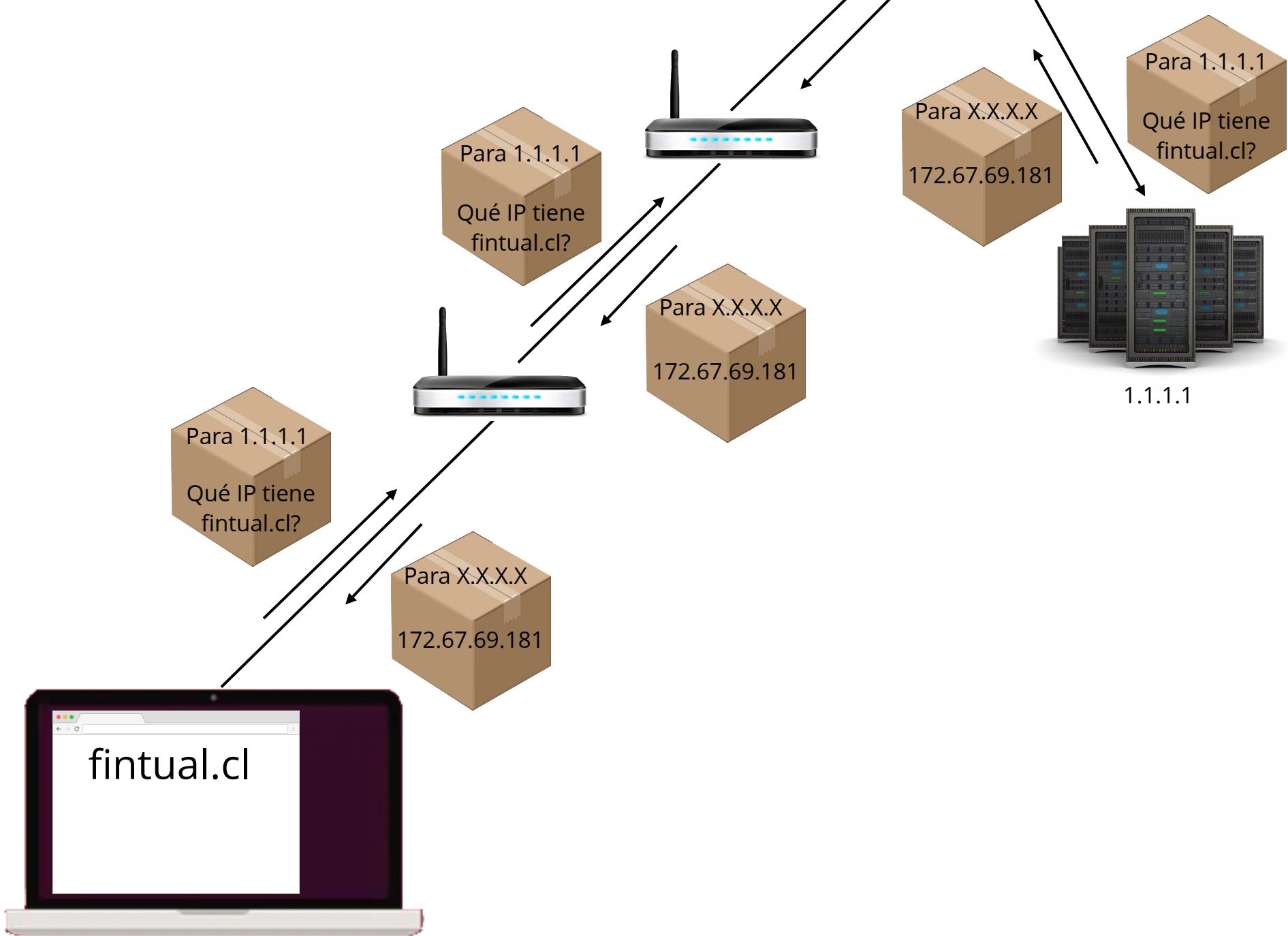


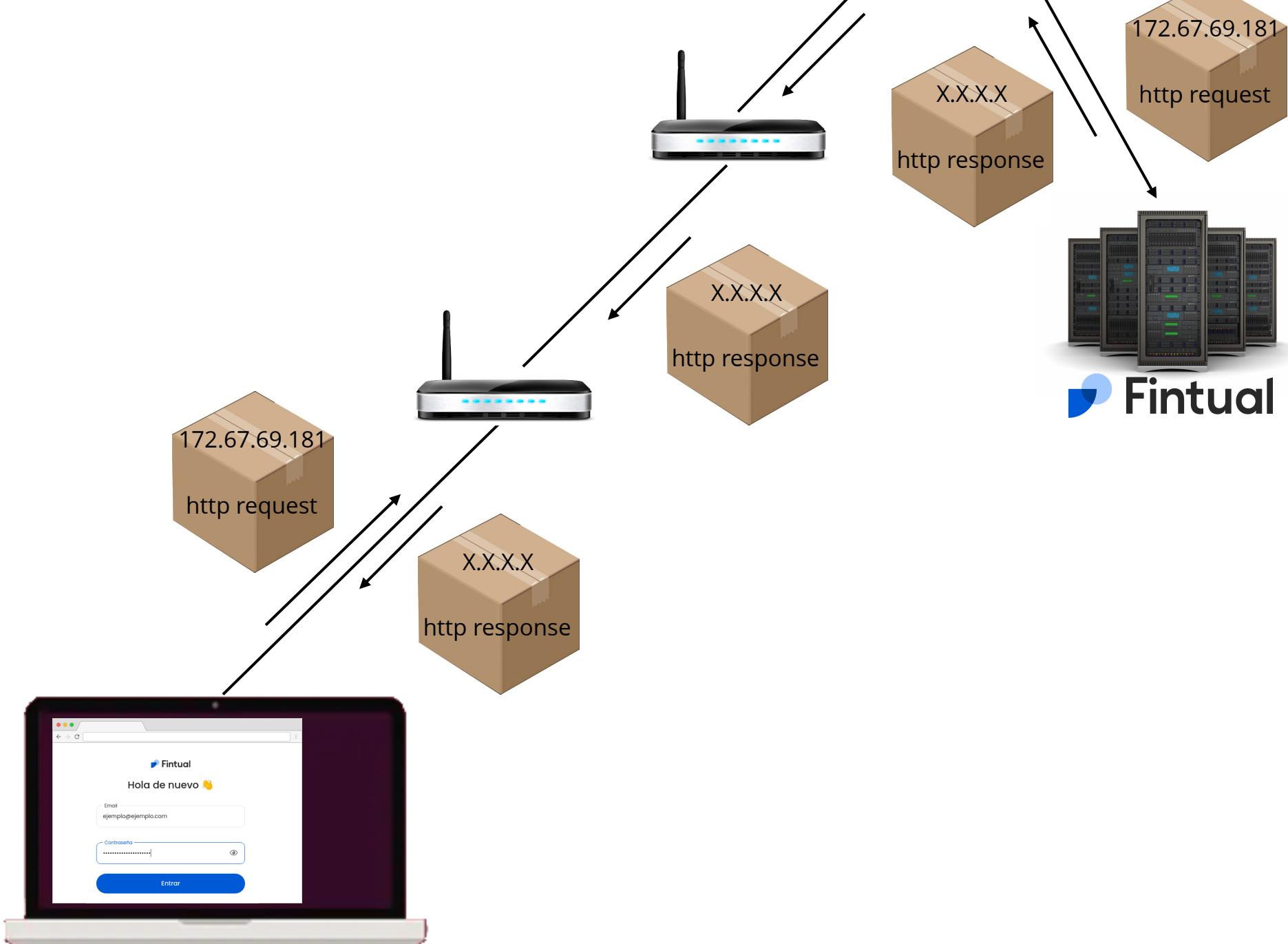


Conozco esa dirección?  
Conozco a alguien que  
conoce esa dirección?  
Ni idea, default route...









**Llegamos...**

**¿Vamos bien?**

**¿Problemas de  
seguridad?**

# IDC 2021 Global DNS Threat Report

## DNS Threat Landscape

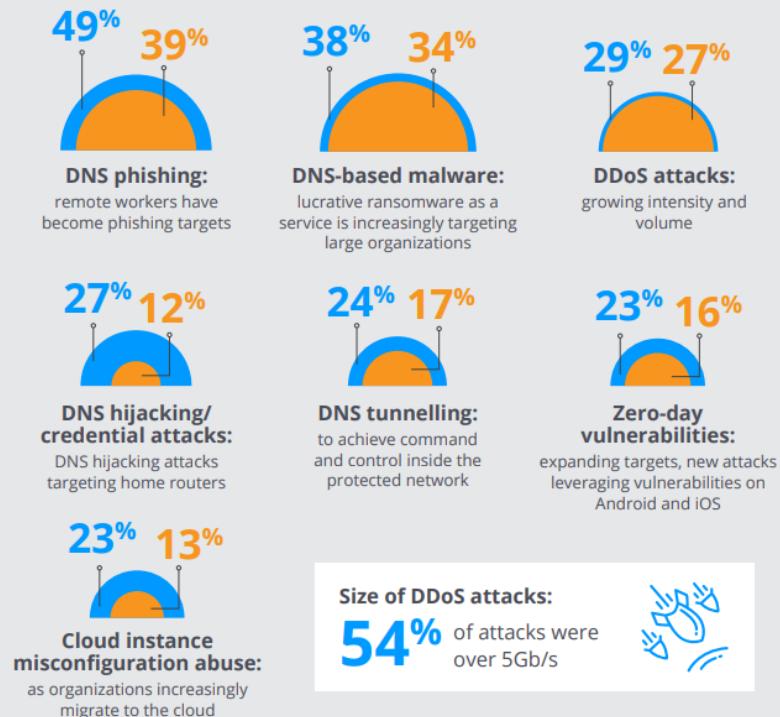
DNS remains a prime target for hackers as it enables them to gain first entry into networks and gain access to data for exfiltration:



With the pandemic rapidly increasing cloud usage and the number of people remote working, the attack surface has increased considerably. As a result, organizations have suffered more diverse types of attacks than ever before, showing that cybercriminals are using all the tools at their disposal to exploit both the DNS protocol and misconfigurations.

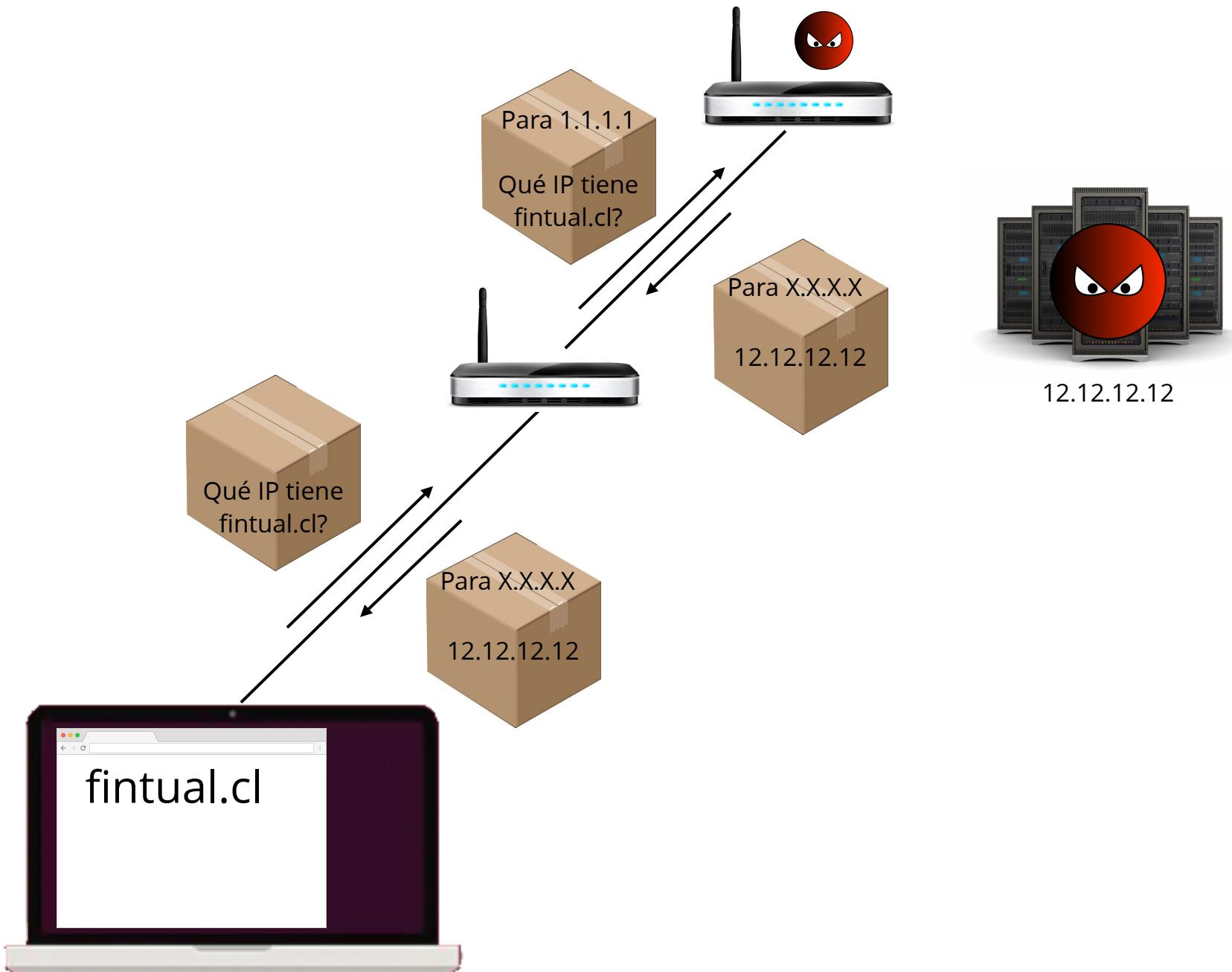
### Top DNS-based attacks:

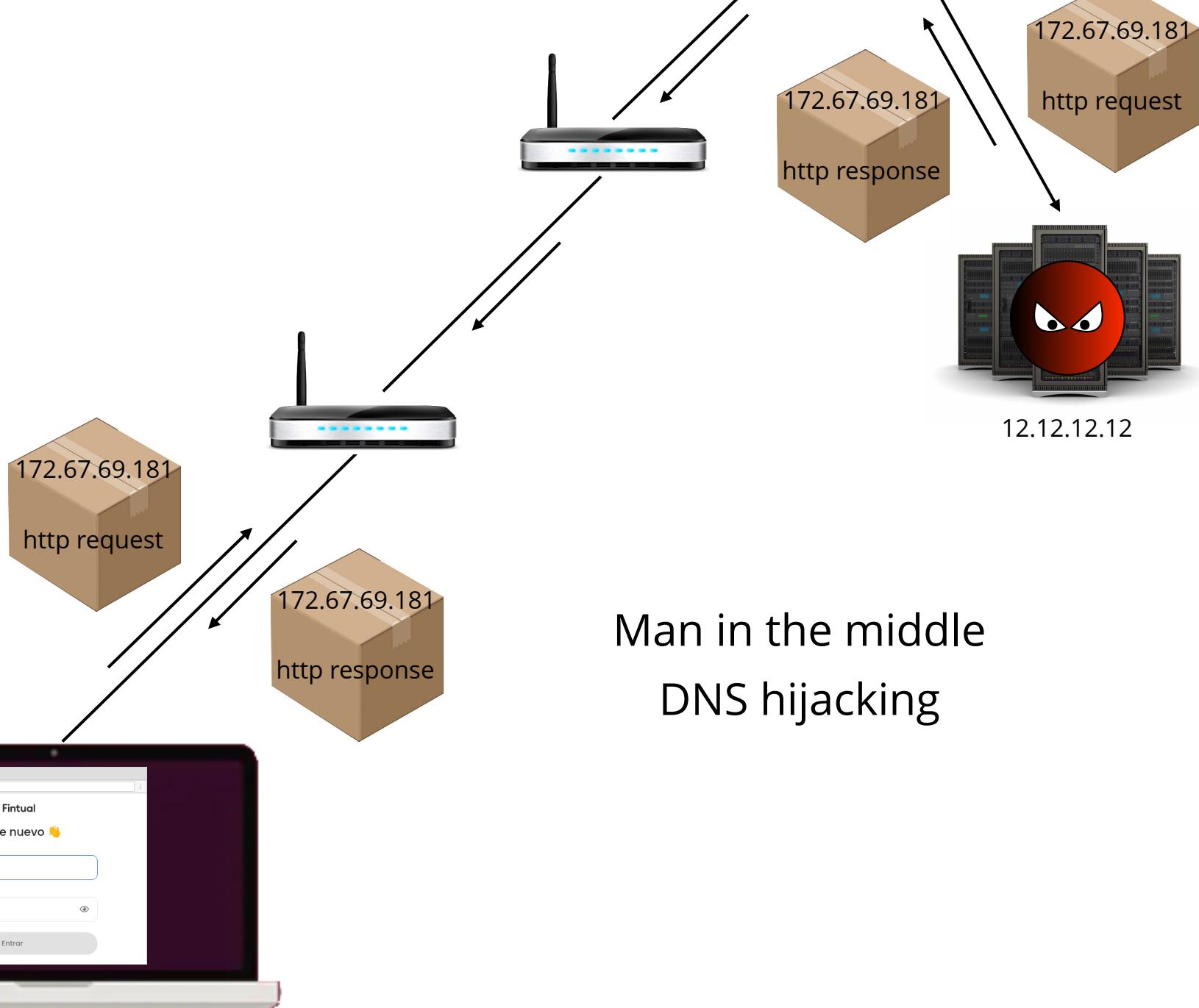
● 2021 ● 2020



The COVID-19 pandemic has created new challenges for businesses as they adapt to WFH operating models. DX initiatives are accelerated, and cybersecurity becomes a major concern:

- Remote workers fall more for phishing scams
- Vulnerabilities leading to credential stuffing and DNS spoofing
- Use of personal devices for work and corporate PC for personal use
- VPNs not user friendly, using too much bandwidth, creating latency leading to poor user experience
- Reliance on home Wi-Fi and home security





# ¿Soluciones?



# Public-key crypto to the rescue





Public key





**¿Problemas?**



# Enter Certificate Authorities (CAs)





Y cómo se yo que  
esta es realmente la  
llave pública de  
Fintual?

Public key



Seguro y autentificado



Firmado por la CA



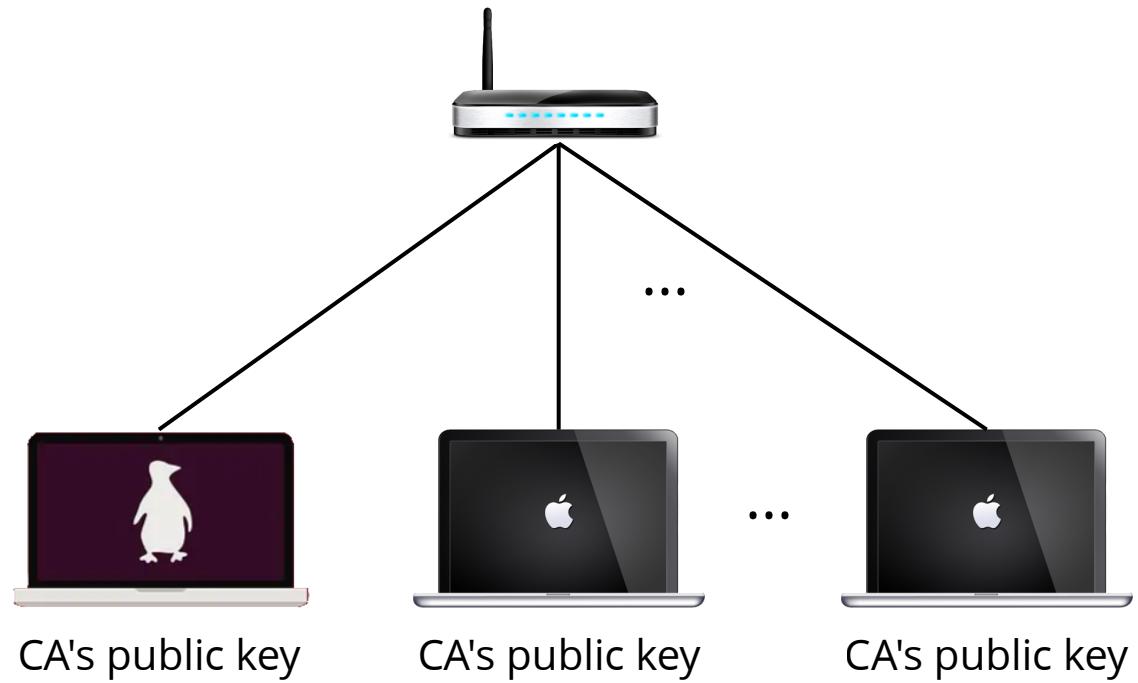
Te dejo un  
certificado  
firmado por la  
CA, dice que esa  
es mi llave  
pública



Private key

**¿Más problemas?**







Y cómo se yo que  
esta es realmente la  
llave pública de  
Fintual?



Te dejo un  
certificado  
firmado por la  
CA, dice que  
esa es mi llave  
pública



 **Fintual**  
Private key

Public key

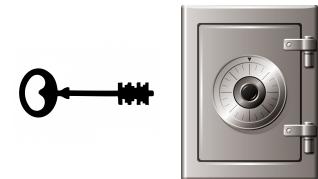
Seguro y autentificado



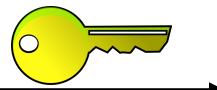
Firmado por la CA

¿Cómo se obtiene este certificado?





Quisiera sacar certificados para fintual.com, aquí mi llave pública



Primero, no te vayas a equivocar, firma <nonce> con la llave privada



Listo!



Además, tienes que poner este <nonce2> en fintual.com/<nonce3>

Todo en orden!

Bacán! me darías un certificado para asegurar a.fintual.com con esta otra llave pública?



Firmado, verificar con

Listo!



Es la llave de  
a.fintual.com

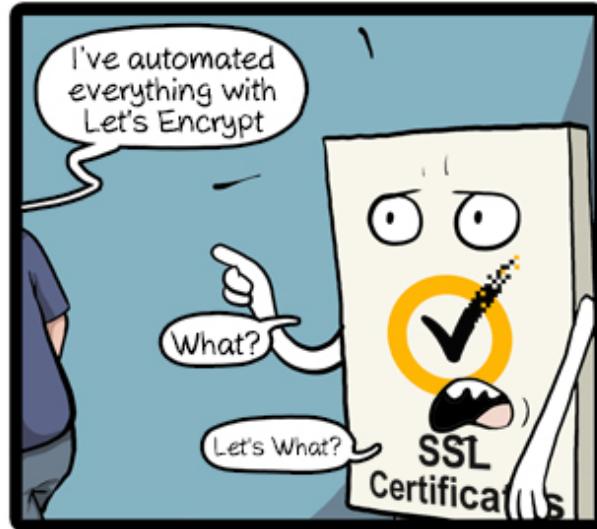
Gracias!

Estos certificados son bastante más complejos en estructura, pero el flujo anterior resume cómo se obtienen

Teniendo certeza de que estoy hablando con el dueño del sitio y tengo la llave pública, aseguramos la sesión con DH

<https://tls12.ulfheim.net/>

para (muchísimos) más detalles







https://fintual.cl



Hola de nuevo 🙌

Email

ejemplo@ejemplo.com

Contraseña

.....

ENCRYPT

ALL THE THINGS



Ruteo IP

DNS

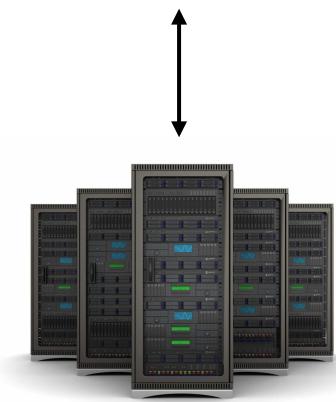
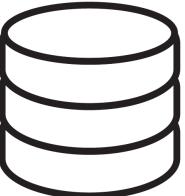
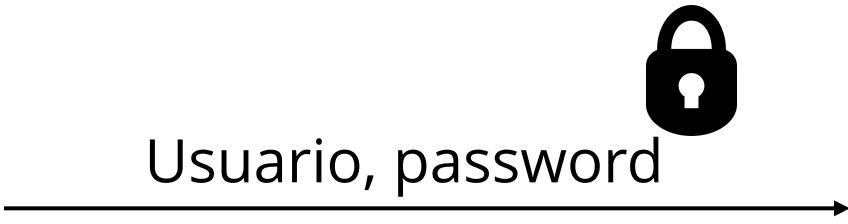
Https

Autoridades Certificadoras

¿Vamos bien?

¿Todavía?





 Fintual

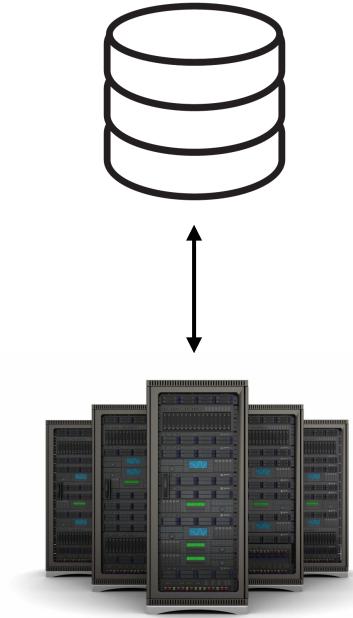
The Fintual logo, which consists of a blue speech bubble-like shape followed by the word "Fintual" in a bold, sans-serif font.

Es cierto que  
este usuario  
tiene este  
password?

A large block of text in Spanish asking if the user has the correct password. The text is aligned to the right and uses a large, clear font.

<b>Correo</b>	<b>Password</b>
atac@ble.com	1Cl4v3muyM4l4
vulner@ble.cl	1Cl4v3P3s1m4

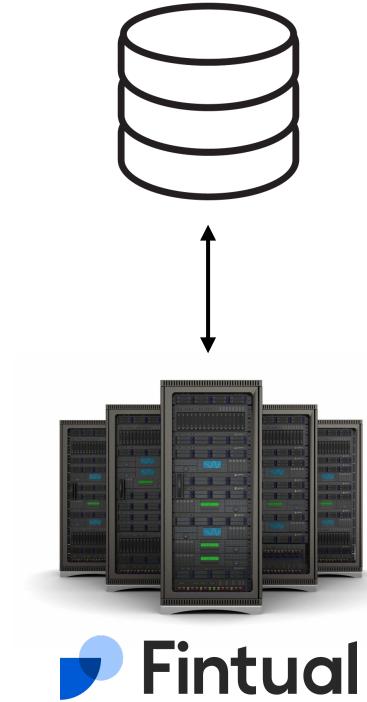
Cualquier persona que gane acceso a la base de datos podría ver correos y contraseñas...



Es cierto que este usuario tiene este password?

**¿Cómo lo arreglamos?**

Correo	SHA256(password)
atac@ble.cl	ef52045429b9094900170095 d91e2bb6b78c514b
vulner@ble.cl	30df6af7eb3dd9e5b9944020 440b9fc6d325beec



Mejor, pero todavía hay problemas...

Es cierto que este usuario tiene este password?

# Rainbow Tables

Las 500.000 Contrasen as m s comunes de 2022

123456	356a192b7913b04c54574d18c28d46e6395428ab
123456789	da4b9237bacccdf19c0760cab7aec4a8359010b0
picture1	77de68daecd823babbb58edb1c8e14d7106e83bb
password	1b6453892473a467d07372d45eb05abc2031647a
12345678	ac3478d69a3c81fa62e60f5c3696165a4e5e6ac4
111111	c1dfd96eea8cc2b62785275bca38ac261256e278
123123	902ba3cda1883801594b6e1b452790cc53948fd
12345	da4b9237bacccdf19c0760cab7aec4a8359010b0
1234567890	77de68daecd823babbb58edb1c8e14d7106e83bb
...	...

<b>Correo</b>	<b>SHA256(password)</b>
atac@ble.cl	ef52045429b9094900170095d91e2 bb6b78c514b
vulner@ble.cl	30df6af7eb3dd9e5b9944020440b9f c6d325beec

¿Hay algún hash de mi rainbow table en la tabla de usuarios?

¿Cómo arreglamos esto?

<b>Correo</b>	<b>Salt</b>	<b>Hash(password    Salt)</b>
c@lidad.cl	fz4/fho#hg%gjs	ef52045429b909490017 0095d91e2bb6b78c514b
mejor@s.cl	zxc\$f4(w@hvg>	30df6af7eb3dd9e5b994 4020440b9fc6d325beec

¿Por qué no podemos atacar con rainbow tables?

Hagámoslo todavía más desagradable para los atacantes!

<b>Correo</b>	<b>HKDF(password)</b>
impec@ble.cl	pbkdf2_sha256\$260000\$W0U57Kqw5UDvasFO 0YQccX\$GAbxNa/PfEtEnS3APi5eV356wpdkfo3ba QNdGsNn2e8=
inmejor@ble.cl	pbkdf2_sha256\$260000\$JxxEv17MH36GQeNao qScmQ\$5vfidwdL96PEWmaZMH3WrVPkoNXoM cO6JrOUKK24vf8=



Es cierto que  
este usuario  
tiene este  
password?

**¡Ganamos acceso!**



https://fintual.cl



Hola de nuevo 

Email

ejemplo@ejemplo.com

Contraseña

.....



Entrar



 <https://fintual.cl>



# Fintual

Hola Martín 

Invirtiendo fácil hace 477 días

 Nuevo Objetivo

 Invertir más

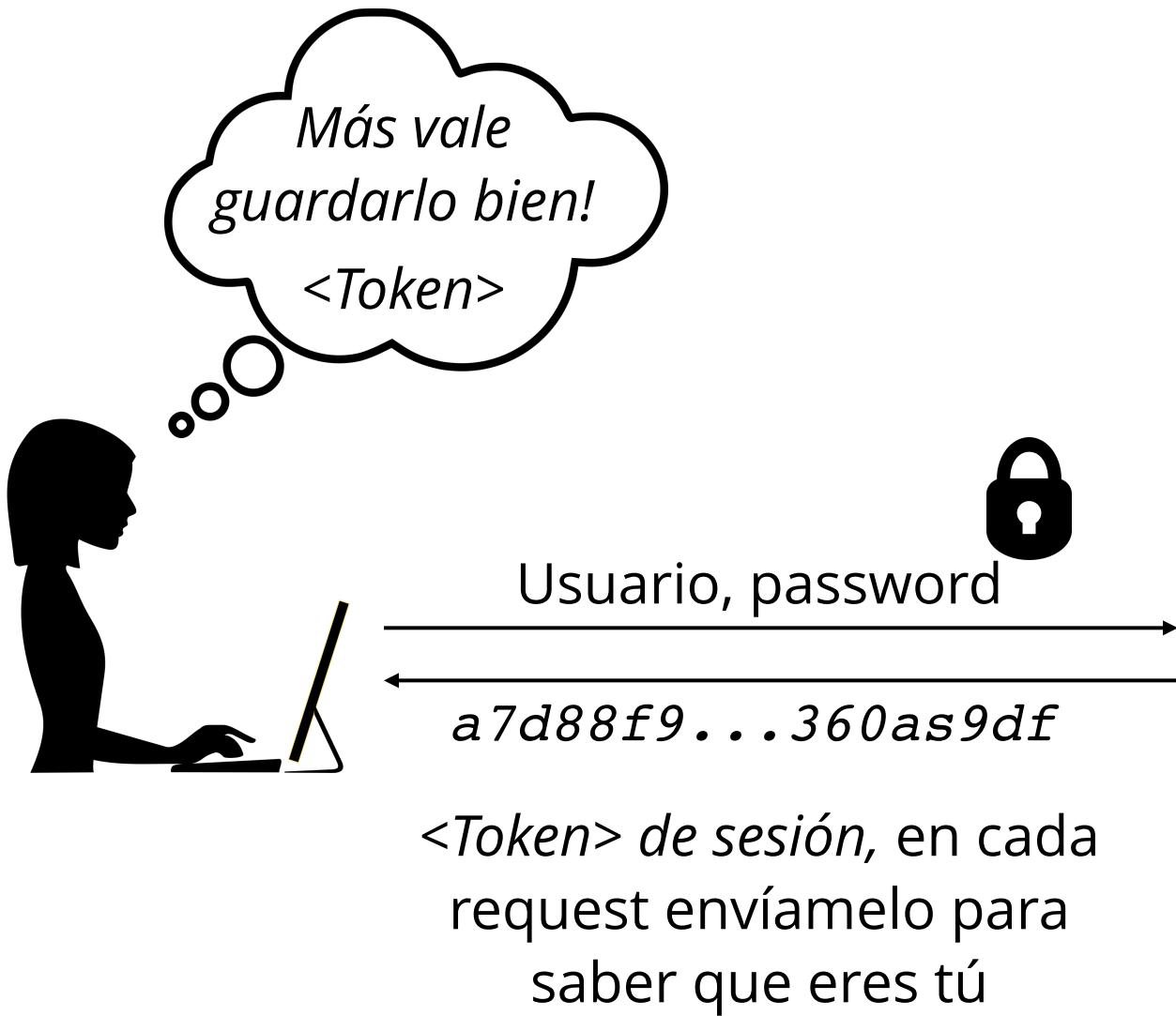




Quiero invertir más



**¿Cómo lo arreglamos?**



# ¿Dónde lo guardamos?

## ¿Opciones?

Variable en JavaScript

LocalStorage

Cookie

Variable en JavaScript

Muy volátil, difícil de saber si se cambia

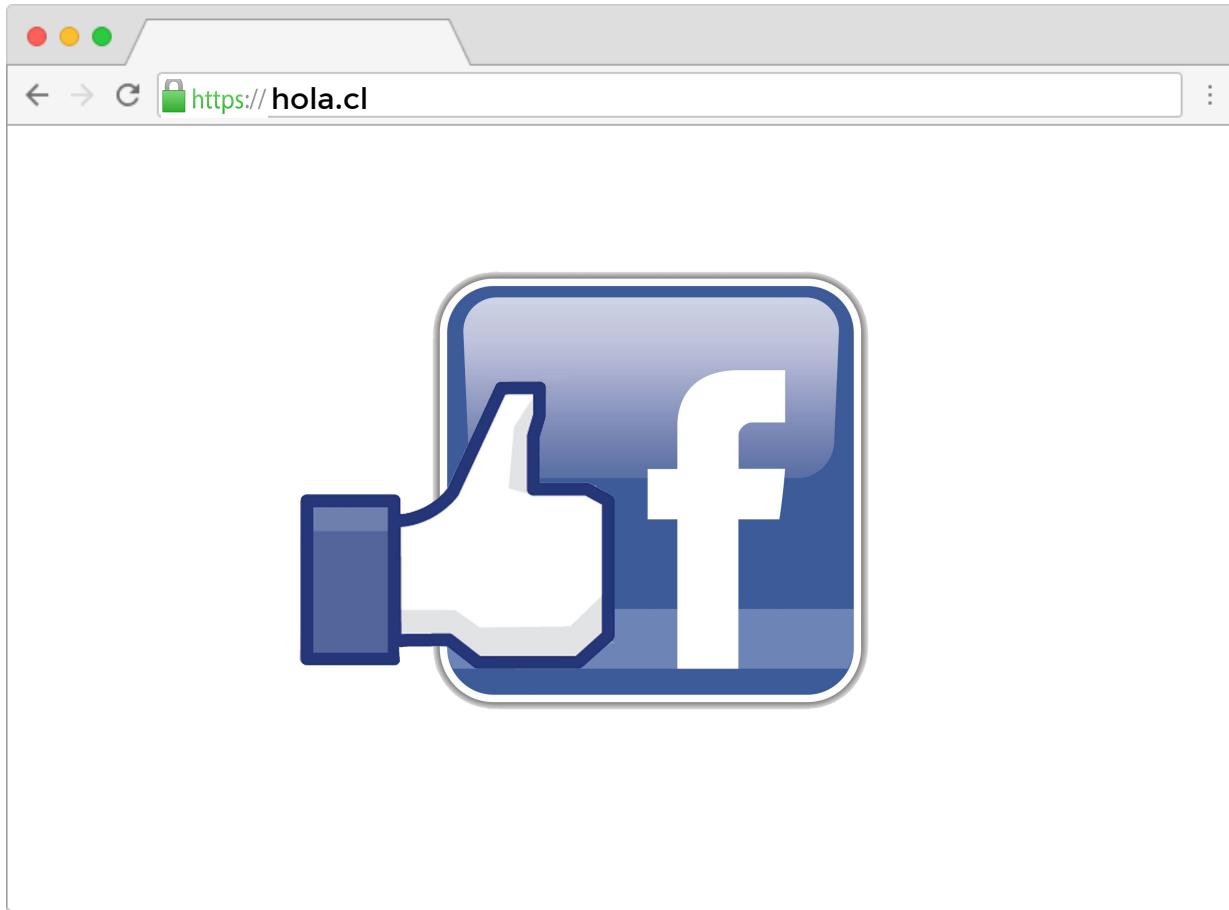
LocalStorage

Podría ser buena idea, ¿pero qué pasa si alguna librería en mi *node\_modules* está infectada?

Cookies

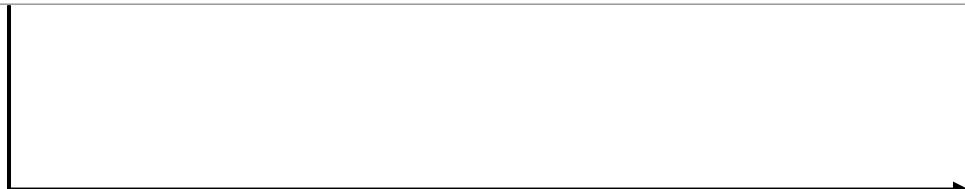
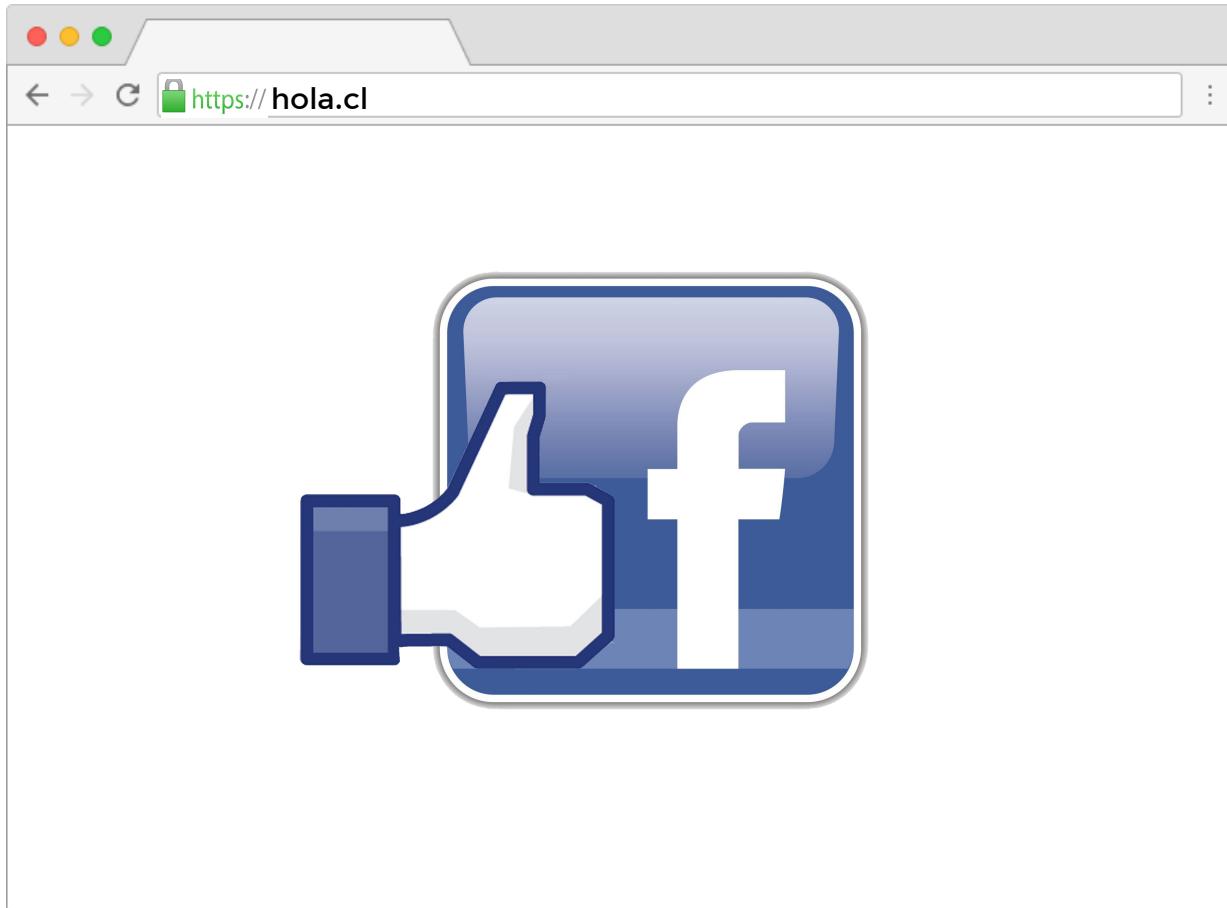
Suena razonable. ¿Qué problema podríamos tener? 

HTTP\_ONLY: Para que no la pueda leer el JS



Le estoy dando un like a hola.cl!!



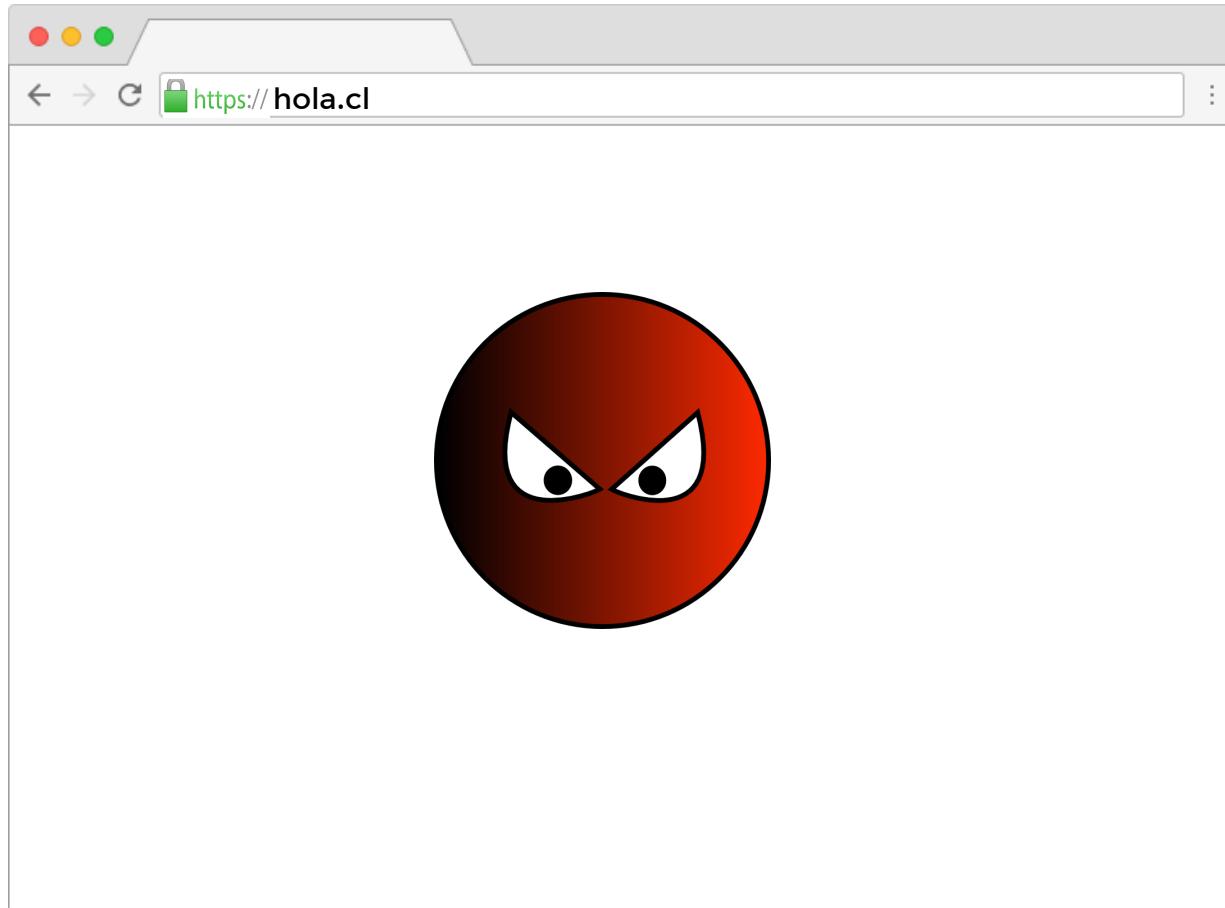


¡Soy yo dando un like!  
(Acá van mis cookies)



¡Perfecto!

**¿Problemas?**



Cross-site  
reference forgery

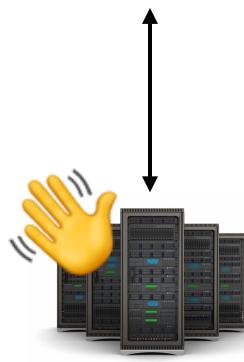
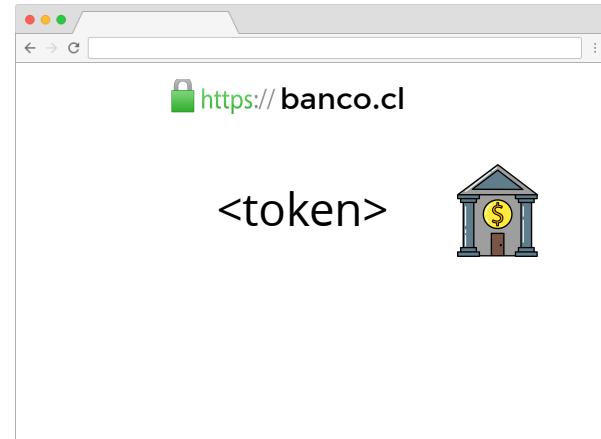
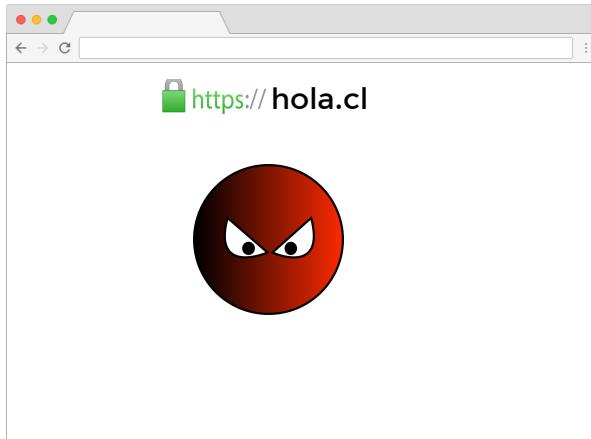


Quiero hacer esta transferencia,  
(Acá van mis cookies)

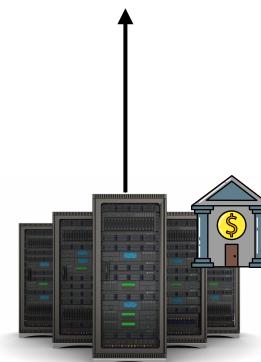


¡Perfecto!





*Quiero transferir  
(Acá van mis cookies)*



user_id	token
1	f2...e4
27	a0...15

Para hacer algo "delicado",  
mándame siempre el token

Cross-site reference forgery token  
A.K.A. **csrf\_token**



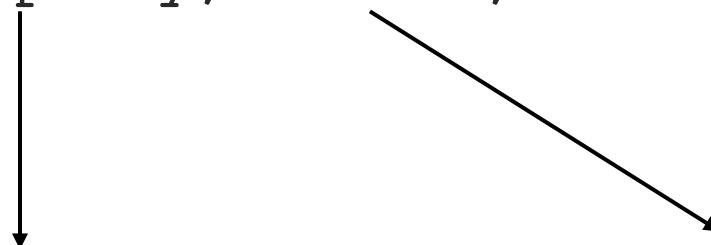
Esta es la forma "tradicional" de mitigar CSRF

Actualmente (desde ~2020) hay mejores formas de protegerse de estos ataques

Set-Cookie: SESSION\_TOKEN=ad94...e10;  
HttpOnly; Secure; SameSite=Strict

Inaccesible por JS

Sólo se envía por HTTPS



# SameSite

El atributo SameSite de una cookie puede tener tres valores

**None:** La cookie se manda siempre (default hasta ~2020)

**Strict:** La cookie se manda sólo si el request se inició en el mismo sitio.

**Lax:** La cookie se manda en requests iniciados por el mismo sitio, y en requests iniciados por otros sitios en los que **cambia la url en la barra de direcciones**





**MUCH TO LEARN,**

**WE ALL STILL HAVE.**

Pero ya tenemos las herramientas necesarias...