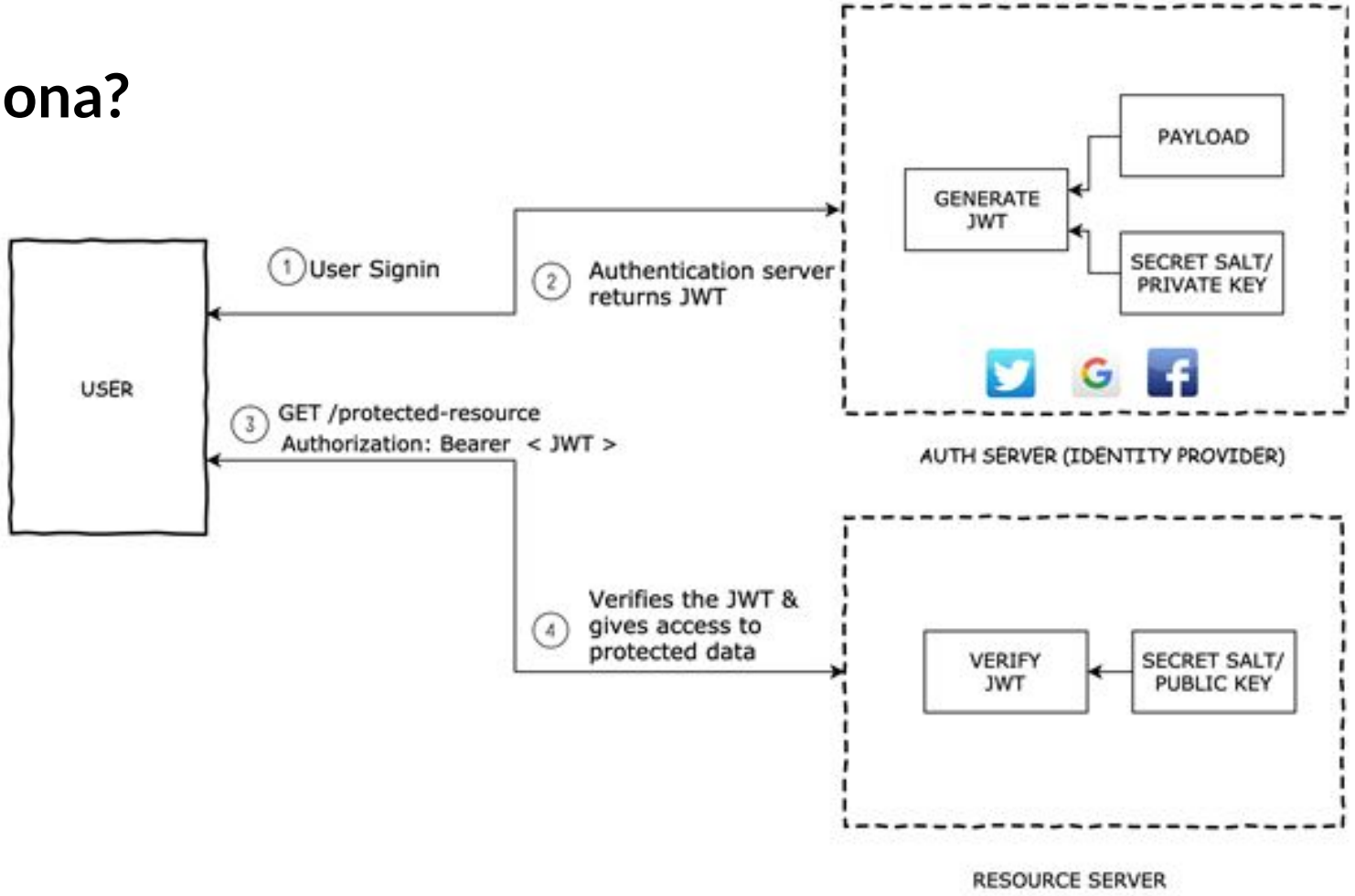
JWT

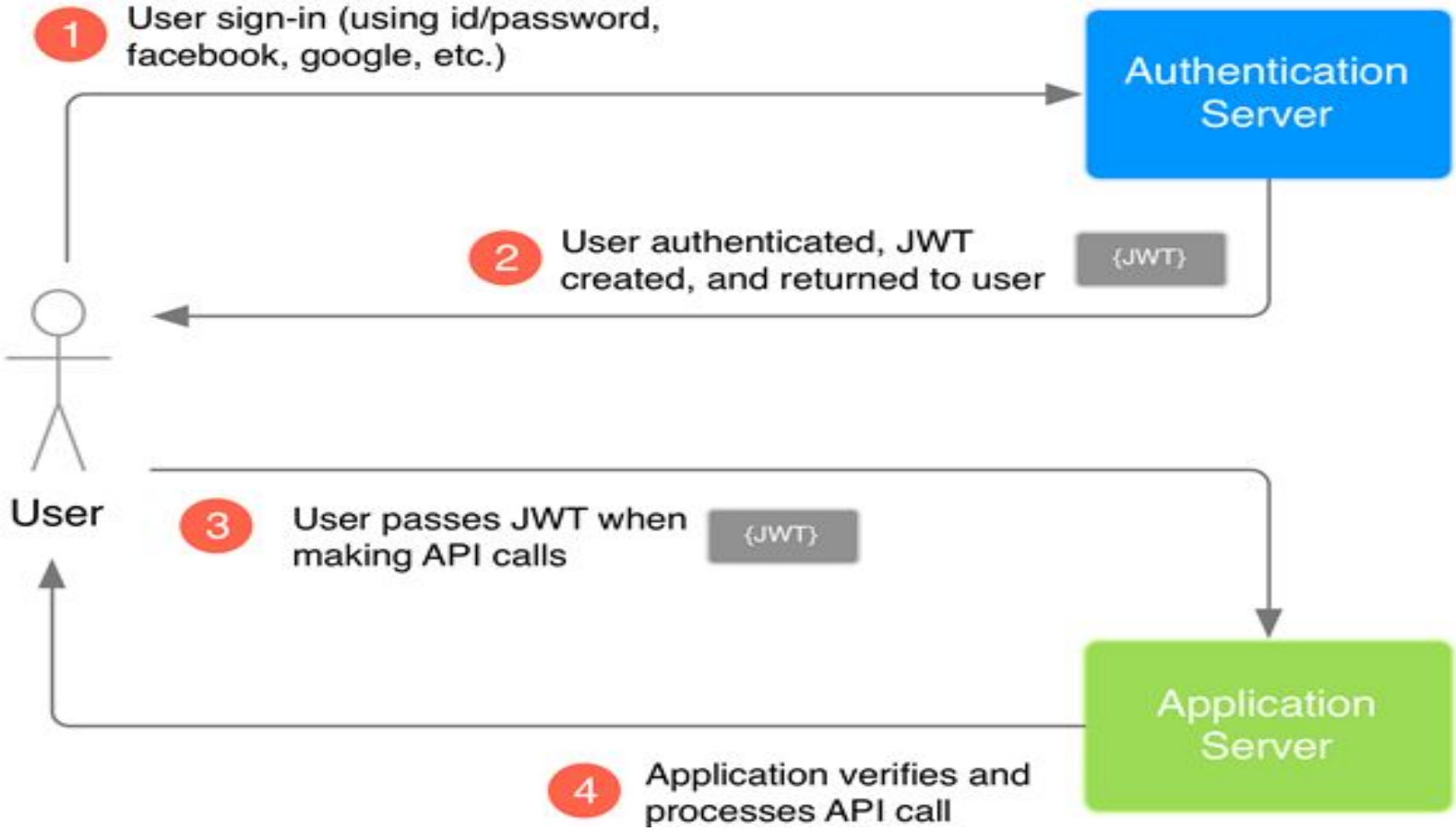
Desarrollo de Aplicaciones en la Nube

Integrantes:
Ulises Ancona
Shaid Bojorquez
Carlos Cuevas

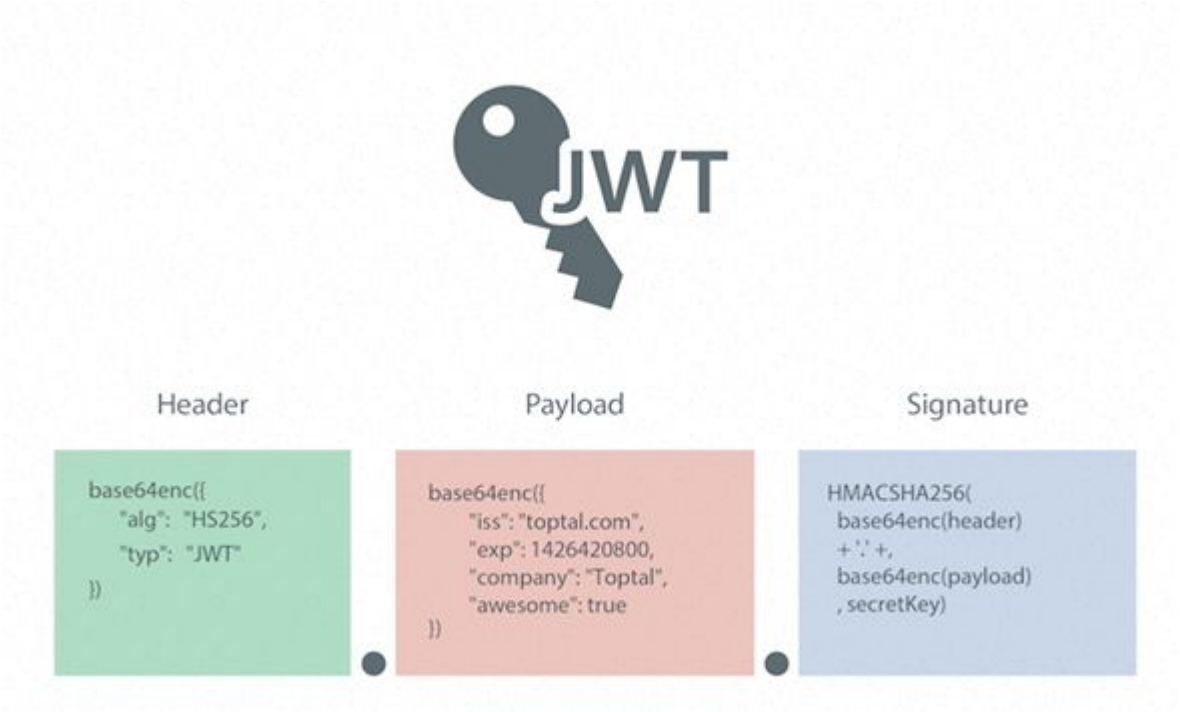
¿Cómo funciona?



¿Cómo funciona?



Estructura del JWT



Signature

Para crear la parte de firma debe tomar el header codificado, el payment codificado, un secreto, el algoritmo especificado en el encabezado y firmarlo.

Por ejemplo, si desea utilizar el algoritmo HMAC SHA256, la firma se creará de la siguiente manera:

```
HMACSHA256(
  base64UrlEncode(header) + "." +
  base64UrlEncode(payload),
  secret)
```

Signature

La firma se utiliza para verificar que el mensaje no se ha cambiado en el camino y, en el caso de los tokens firmados con una clave privada, también puede comprobar que el remitente del JWT es quien dice que es.

“l*a información puede ser verificada y confiable porque está firmada digitalmente*", con un **"secret-key".**

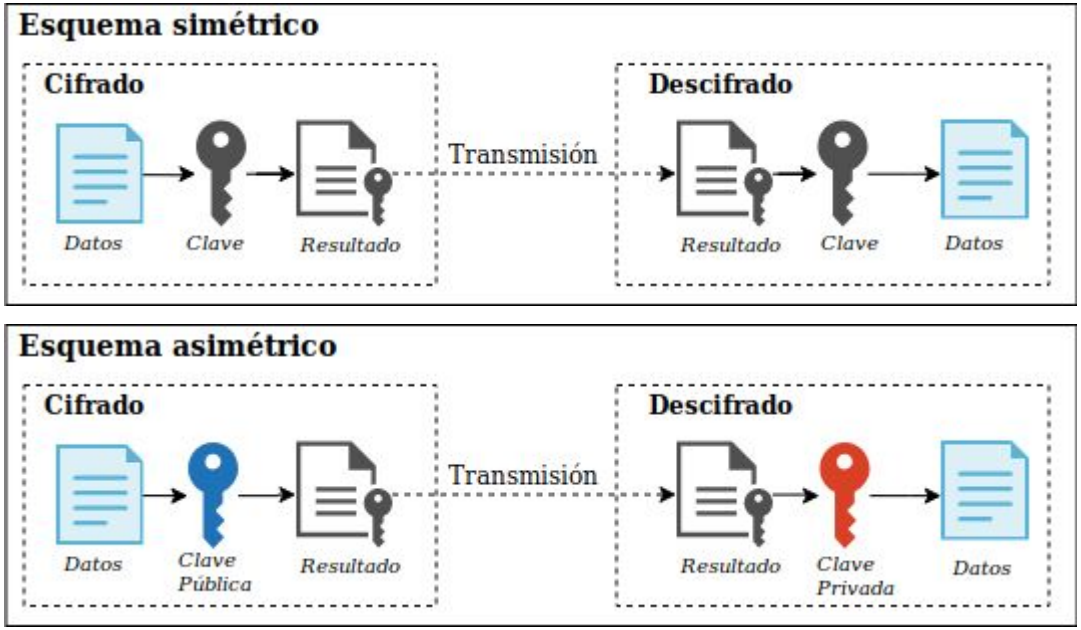
Sólo el lado del servidor debe conocer el “secret key”

Signature

Un JWT se puede cifrar usando una clave simétrica (secreto compartido) o claves asimétricas (la clave privada de un par privado/público).

- Clave simétrica: la misma clave sirve para el cifrado (cuando se crea el JWT) y el descifrado.
- Claves asimétricas: se usan claves diferentes para el cifrado (clave privada) y el descifrado (clave pública).

Signature


Esquema simétrico
Cifrado
Datos
Clave
Resultado
Transmisión
Descifrado
Resultado
Clave
Datos
Esquema asimétrico
Cifrado
Datos
Clave Pública
Resultado
Transmisión
Descifrado
Resultado
Clave Privada
Datos

Ejemplos

Encoded PASTE A TOKEN HERE

```
eyJhbGciOiJFUzM4NCIsInR5cCI6IkpXVCIsImtpZCI6ImlUcVhYSTB6YkFuSkNLRGFvYmZoa00xZi02ck1TcFRmeVpNUnBfMnRLSTgifQ.eyJpZCI6IjEwIiwibm9tYnJlIjoiQ2FybG9zIEN1ZXZhcyIsInVzdWFyaW8iOiJjYXJsb3NAY29ycmVvLm14IiwibGljZW5jaWF0dXJhIjoiTElTIiwiZXF1aXBvSWQiOjcsInBhc3N3b3JkIjoicGFzc3dvcmQifQ.kXWrkoHkMvgw0Mdw8hxcxmqtnZq9Krr4OidIckIZzLJ06OPpJqd7j7ND_eSqKNiegZj-G3Mu2EIQc9TBH42aWeE1S7akemGCPe9yWvRMbf2yGyKd5pJuS6SBJz7HZkNs
```

HEADER: ALGORITHM & TOKEN TYPE

```
{
  "alg": "ES384",
  "typ": "JWT",
  "kid": "iTqXXI0zbAnJCKDaobfhkM1f-6rMSpTfyZMRp_2tKI8"
}
```

PAYLOAD: DATA

```
  "id": "10",
  "nombre": "Carlos Cuevas",
  "usuario": "carlos@correo.mx",
  "licenciatura": "LIS",
  "equipoId": 7,
  "password": "password"
}
```

VERIFY SIGNATURE

```
ECDSASHA384(
  base64UrlEncode(header) + "." +
  base64UrlEncode(payload),
  -----BEGIN PUBLIC KEY-----
  MHYwEAYHKoZIzj0CAQYFK4EEACID
  YgAEC1uWSXj2czCDwMTLWV5BFmwx
  dM6PX9p+
  ,
  -----BEGIN EC PRIVATE KEY---
  --
  MIGkAgEBBDCAHpFQ62QnGCEvYh/p
  E9QmR1C9aLcDItRbslbmhen/h1tt
  8AyMhske
)
```

Encoded PASTE A TOKEN HERE

```
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCIsImt
pZCI6ImlUcVhYSTB6YkFuSkNLRGFvYmZoa00xZi
02ck1TcFRmeVpNUnBfMnRLSTgifQ.eyJpZCI6Ij
EwIiwibm9tYnJlIjoiQ2FybG9zIEN1ZXZhcyIsI
nVzdWFyaW8iOiJjYXJsb3NAY29ycmVvLm14Iiwi
bGljZW5jaWF0dXJhIjoiTElTIiwiZXF1aXBvSWQ
iOjcsInBhc3N3b3JkIjoicGFzc3dvcmQifQ.u_T
H-
z6uCjBghgX_wcRYrY1DSejjaqRHJm5Am1Gd8B8
```

HEADER: ALGORITHM & TOKEN TYPE

```
{
  "alg": "HS256",
  "typ": "JWT",
  "kid": "iTqXXI0zbAnJCKDaobfhkM1f-6rMSpTfyZMRp_2tKI8"
}
```

PAYLOAD: DATA

```
{
  "id": "10",
  "nombre": "Carlos Cuevas",
  "usuario": "carlos@correo.mx",
  "licenciatura": "LIS",
  "equipoId": 7,
  "password": "password"
}
```

VERIFY SIGNATURE

```
HMACSHA256(
  base64UrlEncode(header) + "." +
  base64UrlEncode(payload),
  muysecreto
) ☐ secret base64 encoded
```

Encoded PASTE A TOKEN HERE

eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWIiOiIxMjM0NTY3ODkwIiwibmFtZSI6IlVsaXNlcyBEb2UiLCJhZG1pbiI6dHJ1ZX0.0UWvkEdawMPUBNTh37VwR5a6HCf5gA7t6UID8p4IZNc

Decoded EDIT THE PAYLOAD AND SECRET

HEADER: ALGORITHM & TOKEN TYPE

```
{
  "alg": "HS256",
  "typ": "JWT"
}
```

PAYLOAD: DATA

```
{
  "sub": "1234567890",
  "name": "Ulises Doe",
  "admin": true
}
```

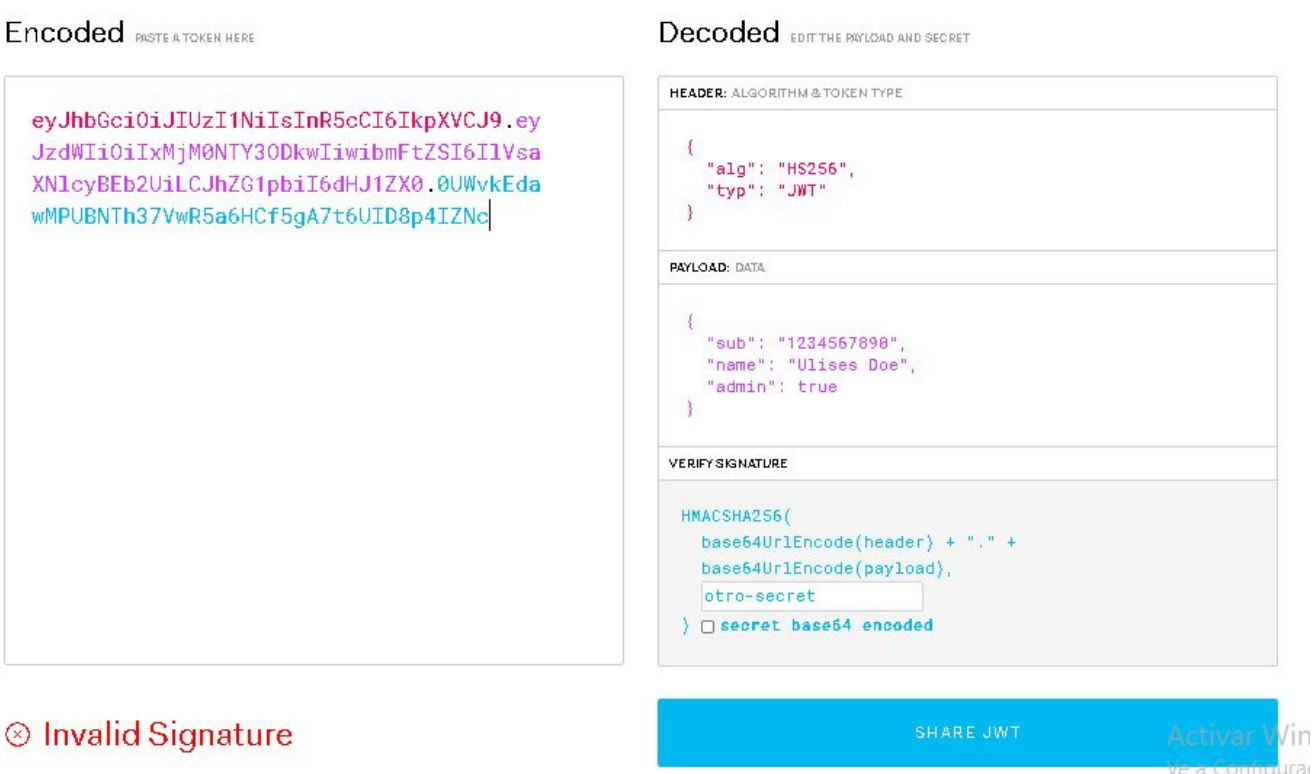
VERIFY SIGNATURE

```
HMACSHA256(
  base64UrlEncode(header) + "." +
  base64UrlEncode(payload),
  secret
) ☐ secret base64 encoded
```

Signature Verified

SHARE JWT

Firma Inválida



Por que es conveniente usarlo

- Como JSON es menos detallado que XML, cuando está codificado, su tamaño también es más pequeño, lo que hace que JWT sea más compacto que SAML.
- Los tokens JWT y SAML pueden usar un par de claves pública / privada en forma de certificado X.509 para firmar.
- Es muy simple firmar el formato JSON.
- Los analizadores JSON son comunes en la mayoría de los lenguajes de programación porque se asignan directamente a los objetos. Por el contrario, XML no tiene un mapeo natural de documento a objeto. Esto hace que sea más fácil trabajar con JWT que con las aserciones SAML.
- En cuanto al uso, JWT se utiliza a escala de Internet.