

BIENVENIDOS
AL CURSO:

Especialización ASP.NET Core 5 Developer

SESIÓN 05





01

ASP.NET Core 5 Identity

02

Implementando el Log in.

03

Implementando el Log out.

04

Implementando registro de usuarios.

05

Implementando cambio de clave (encriptación).

ÍNDICE



Autenticación



Nombre: Erick Aróstegui

Fecha de nacimiento: 17/11/1981



Autenticación

Autenticación

- Determinar la identidad
- Es necesario la comprobación
- La comprobación de una solicitud se hace mediante una clave.
- Si la comprobación es exitosa, se emiten los Claims.

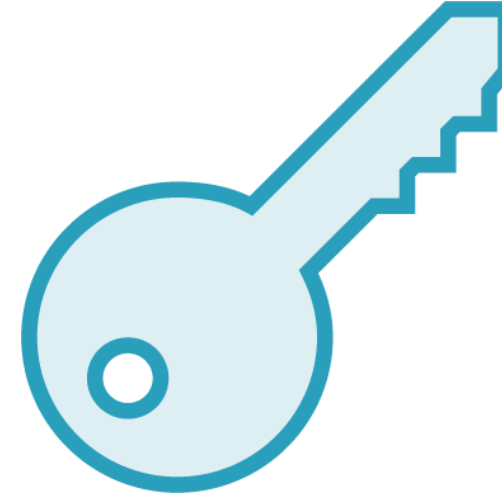


Overview



Authentication

Cookie
Identity
Identity Provider



Authorization

ASP.NET Core
Authorization

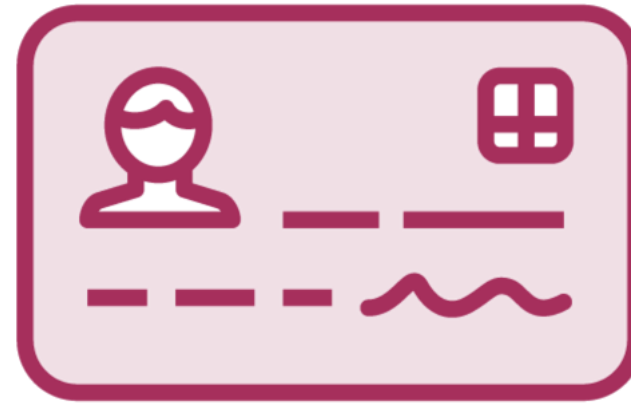


Autorización

Autorización

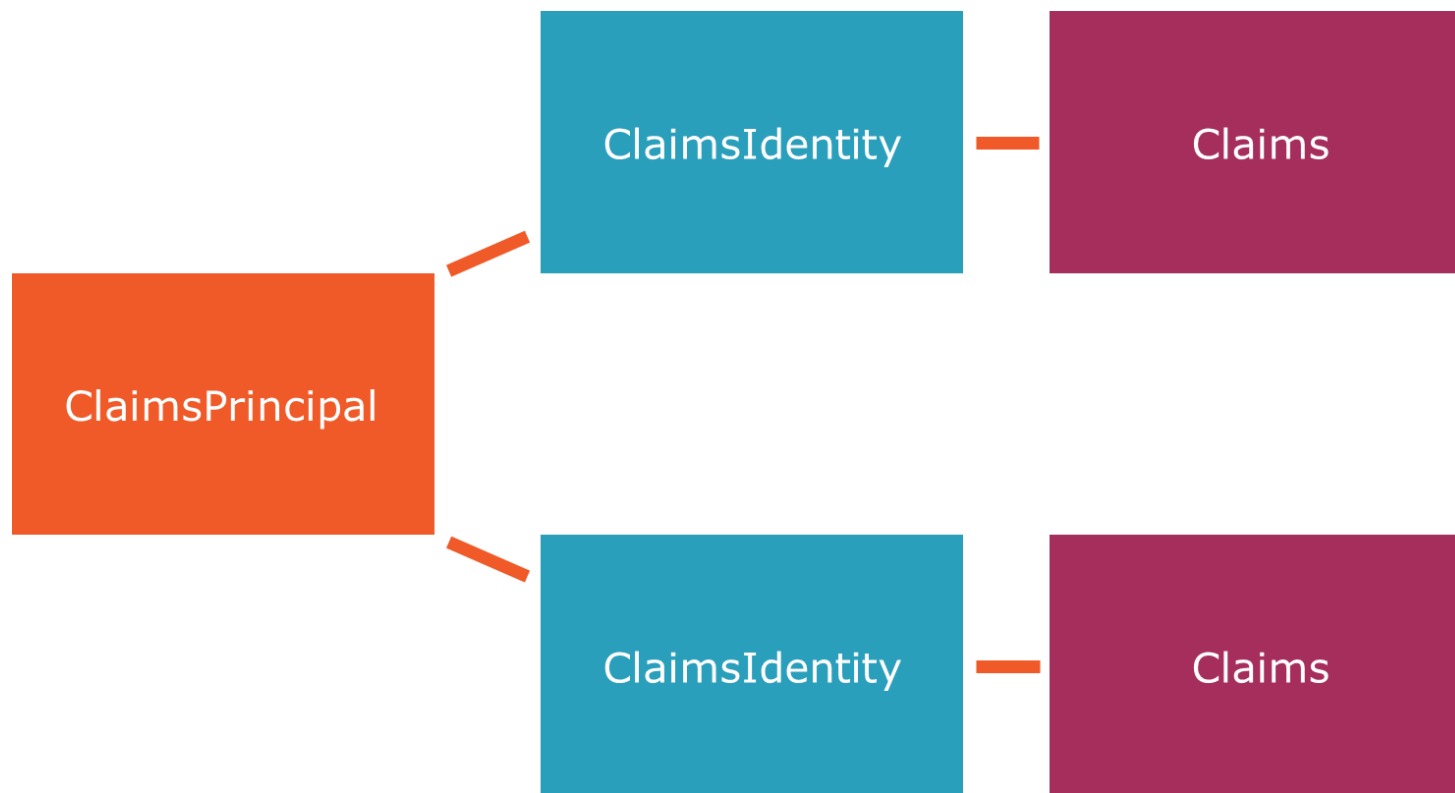
- Limitar el acceso
- Qué acciones puede tomar un usuario
- Necesita autenticación primero

Esquemas de Autenticación





Claims-based Identity





Identity Cookie





Identity Cookie

Identity Cookie

- Rastrea a qué usuario pertenece una solicitud
- Almacena de forma segura la información del usuario
- Cifrado simétrico, clave solo en el servidor
- Se usa para reconstruir el objeto ClaimsPrincipal en cada solicitud
- Asegurado por ASP.NET Core Data Protection



Identity Cookie

Problemas con Identity Cookies

- Las Cookies persistenten de por vida
- El usuario tiene acceso a la aplicación mientras la cookie viva
- Solución: reaccionar ante un evento que se dispara en cada solicitud entrante con una cookie



Identity Cookie



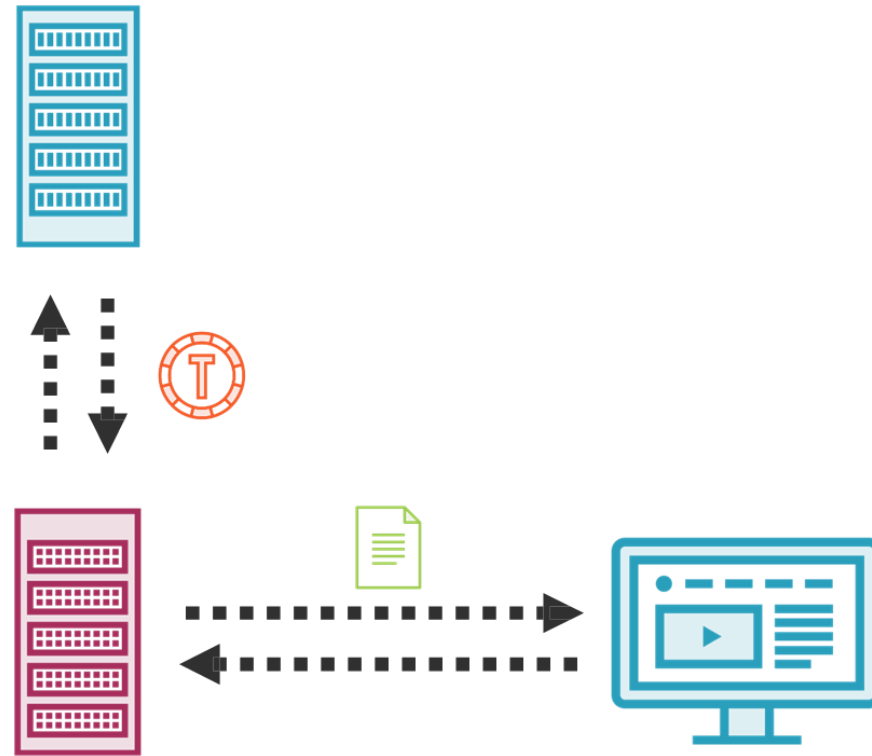


Proveedores de identidad externos

Proveedores de identidad externos

- Google
- Facebook
- Microsoft
- Twitter

Proveedores de identidad externos





Google Provider

<https://4sh.nl/GoogleAddApp>



Proveedores de identidad externos

Scheme Actions

- Authenticate
- Challenge
- Forbid



Identity

Que es Identity?

- Framework de autenticación
- Contiene clases auxiliares y UI
- Personalizable
- Configurable



Identity

Características

- Login y logout
- Registro de usuarios
- Logins con terceros
- Gestion de contraseñas
- Bloqueo de cuentas
- Two-factor authentication



Identity

Identity Configuration

- Valores predeterminados configurables



Identity

Identity's UI Personalizable

- ¿Necesario para todas las páginas?
- `_ViewStart.cshtml` ya establece la página de diseño temático
- Algunas páginas necesitan más personalización que otras



Identity

Identity's UI

- Login/logout
- Confirmación de dirección de correo
- Gestion de contraseñas
- Bloqueo de cuenta
- Two factor authentication
- Personal data



DbContext Options

DbContext

- Derivar DbContext existente de IdentityDbContext
- Use un DbContext separado para Identity apuntando a la misma base de datos
- Use un DbContext separado para Identity usando diferentes bases de datos



ASP.NET Core Identity

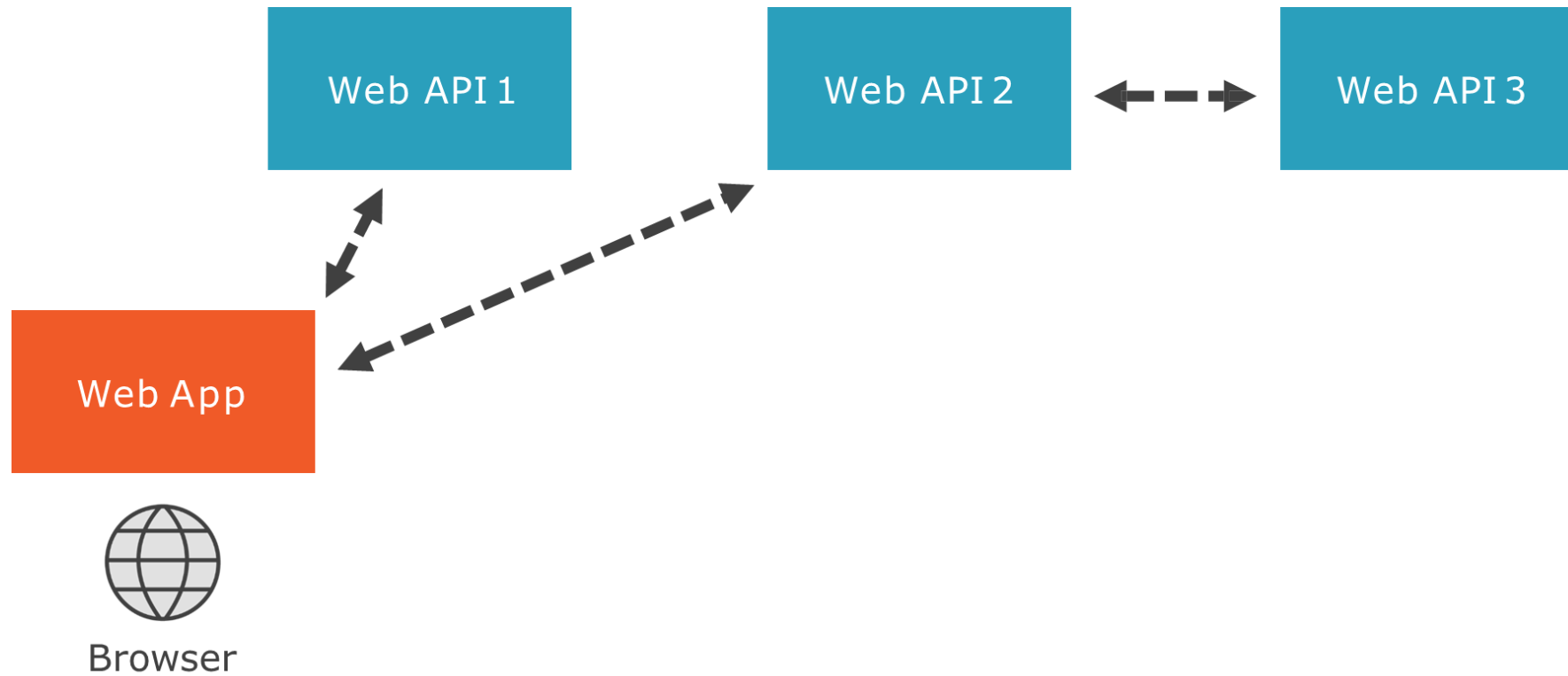


Roles

El soporte de roles está desactivado,
debe activarse explícitamente.



Vista de aplicación





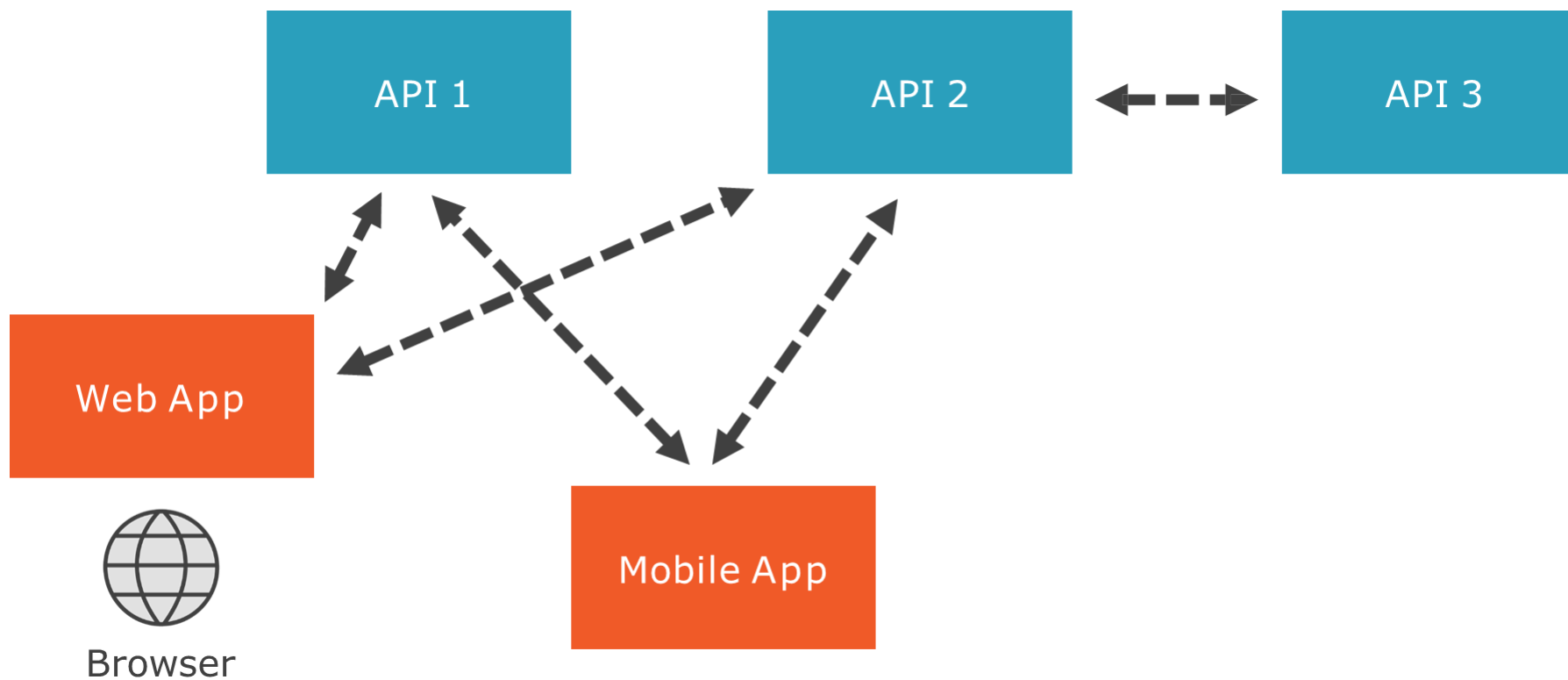
Vista de aplicación

Identity en una aplicación

- Las cookies son para una URL
- ¿Usar un truco o inventar el tuyo?
- No. Muy difícil hacerlo seguro.
- Usar OpenIdConnect

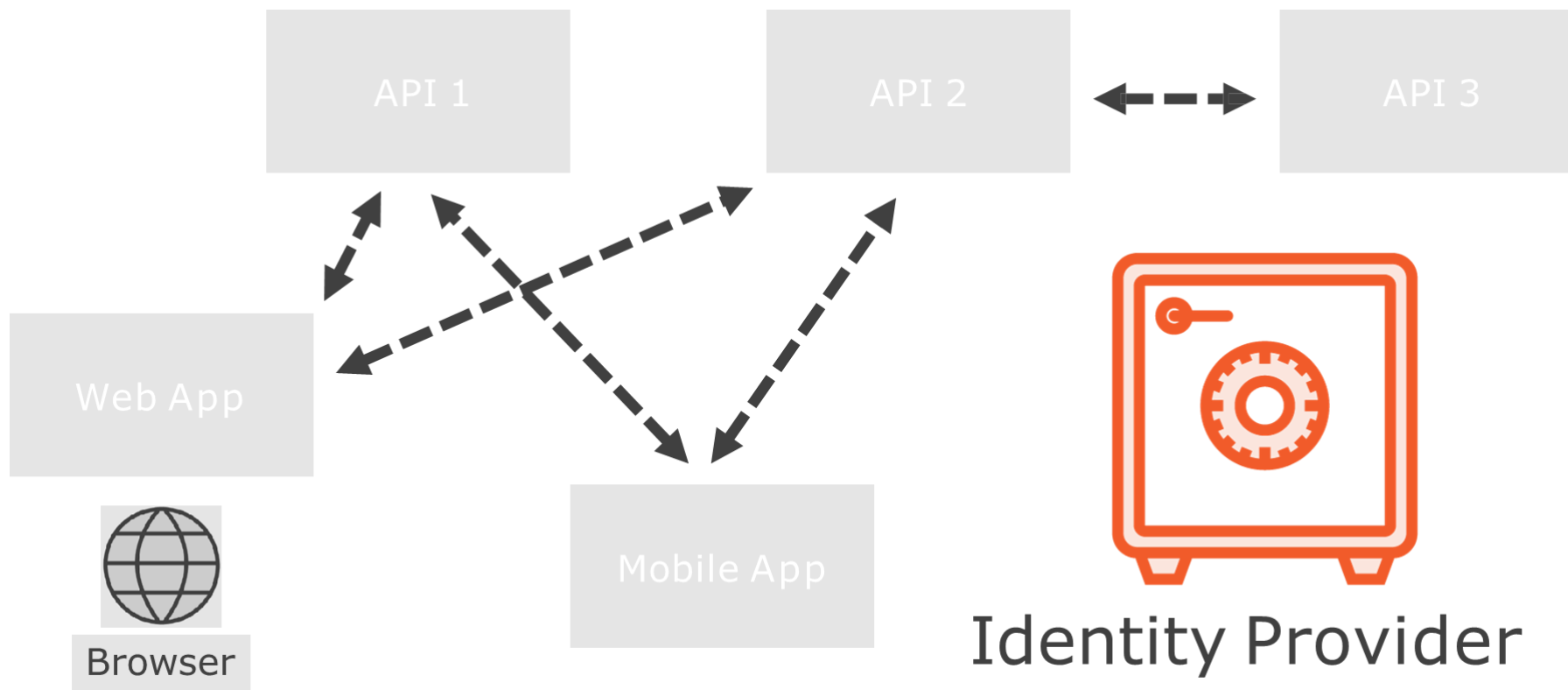


Vista de una aplicación típica





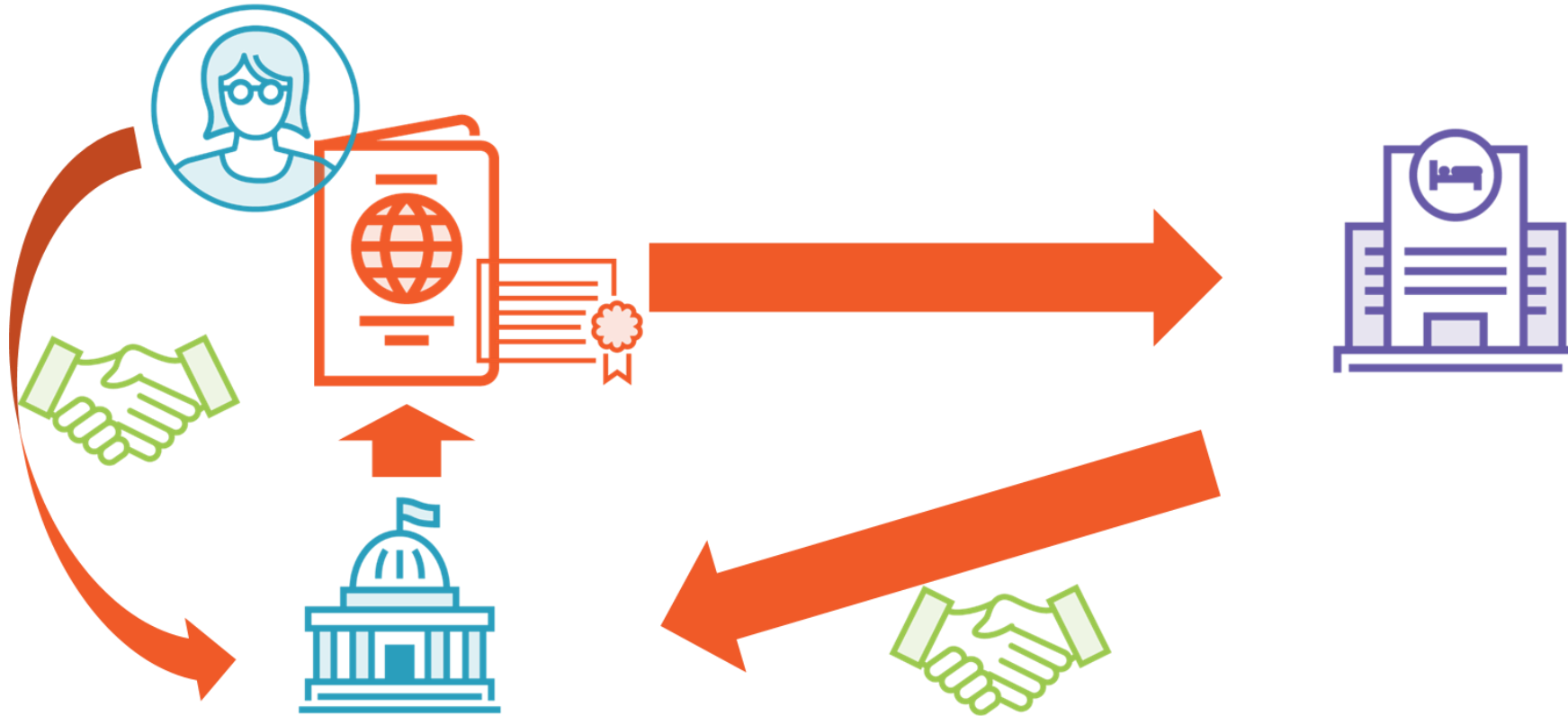
Identity Provider



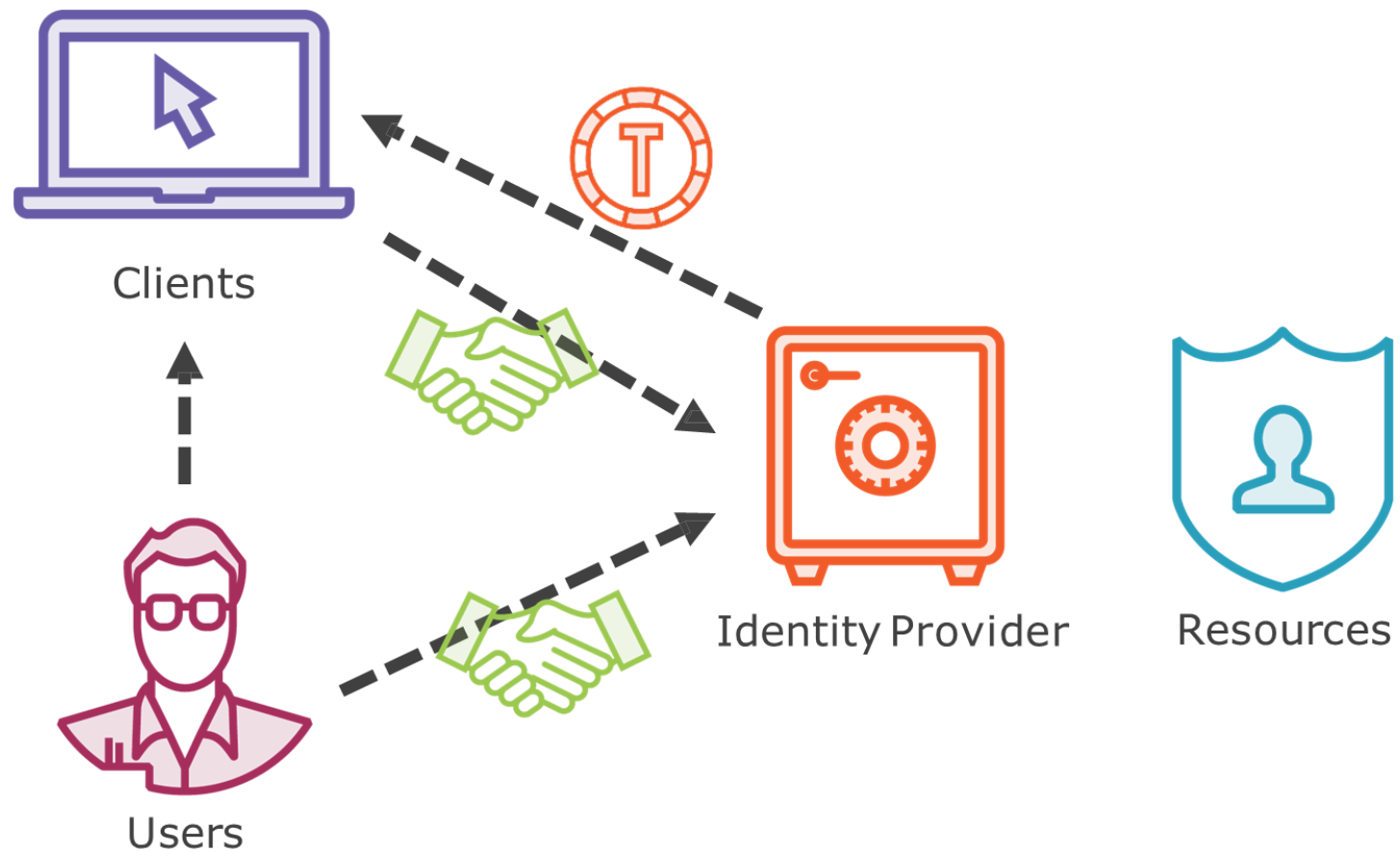
Un proveedor de identidad para gobernarlos a todos



El proceso de autenticación mejorado

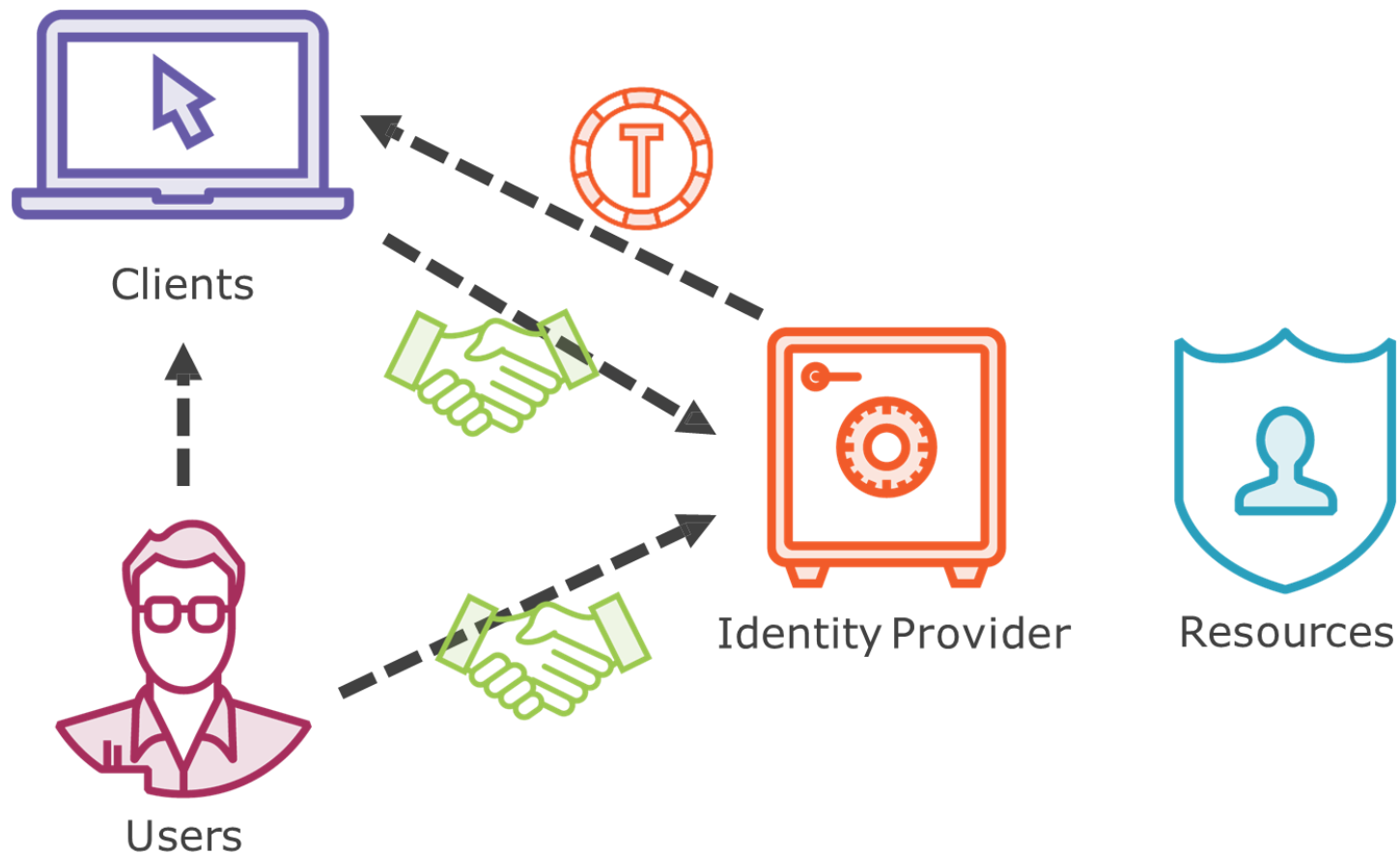


Conceptos



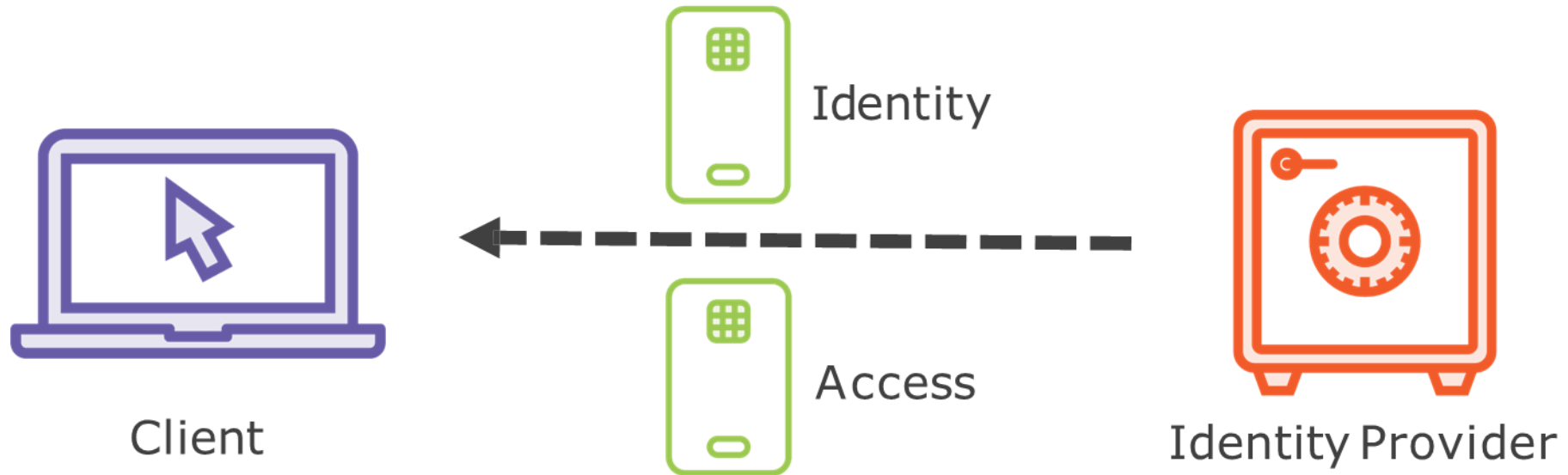


Conceptos



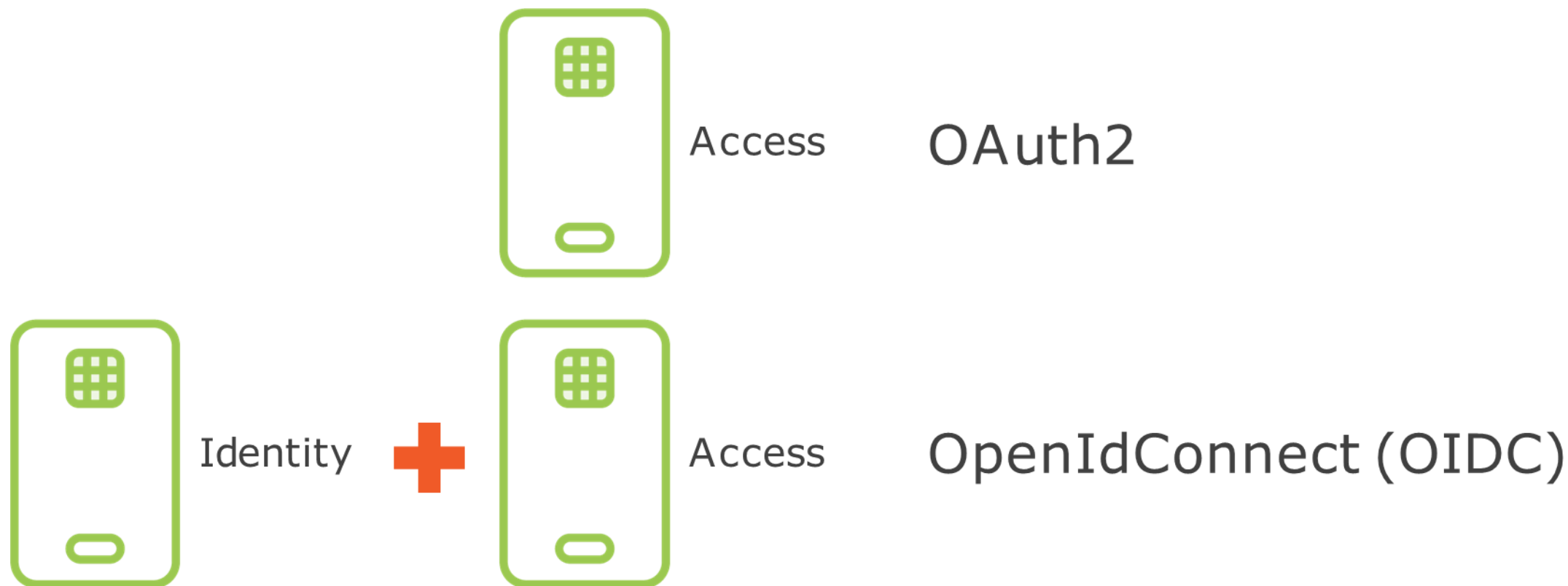


Tokens



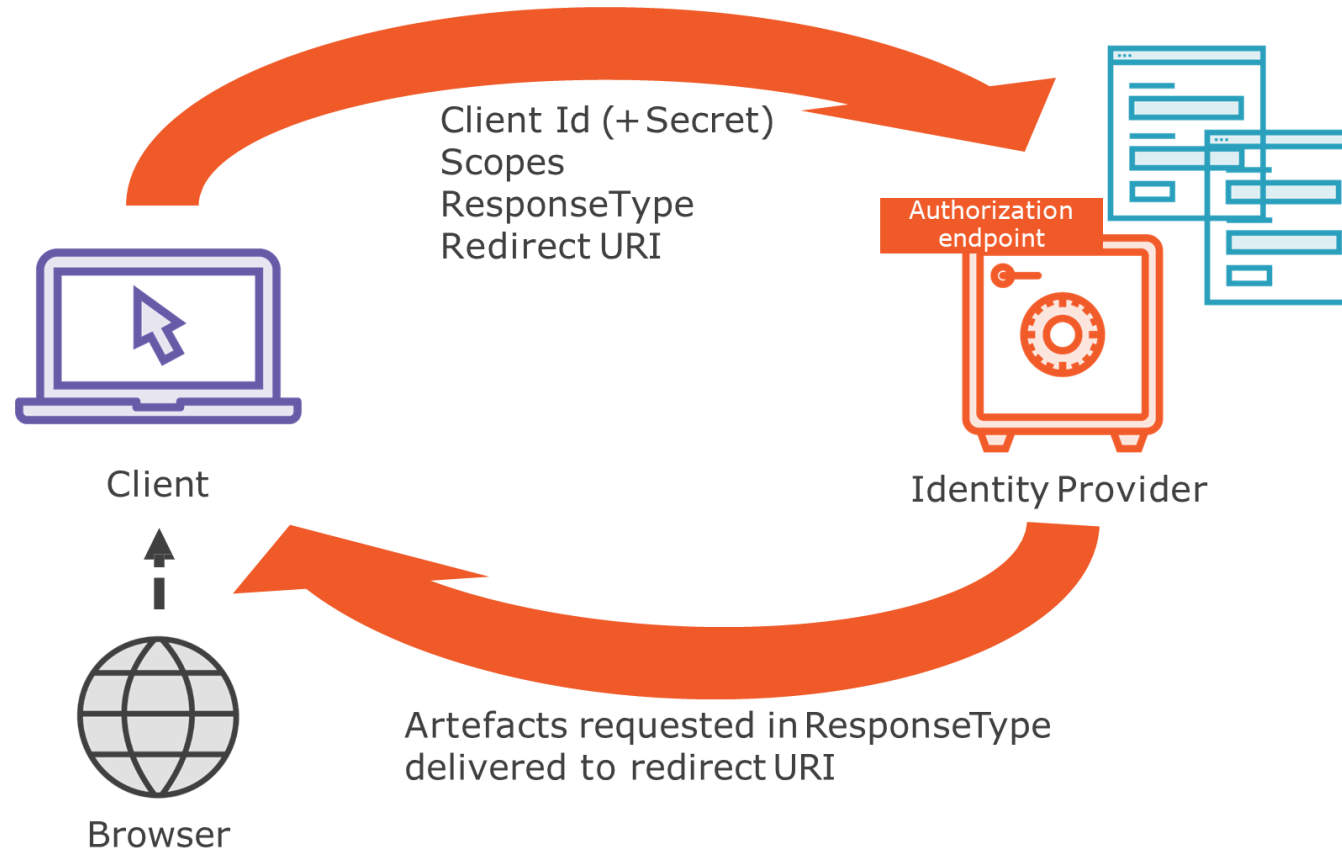


Estandares





Estandares





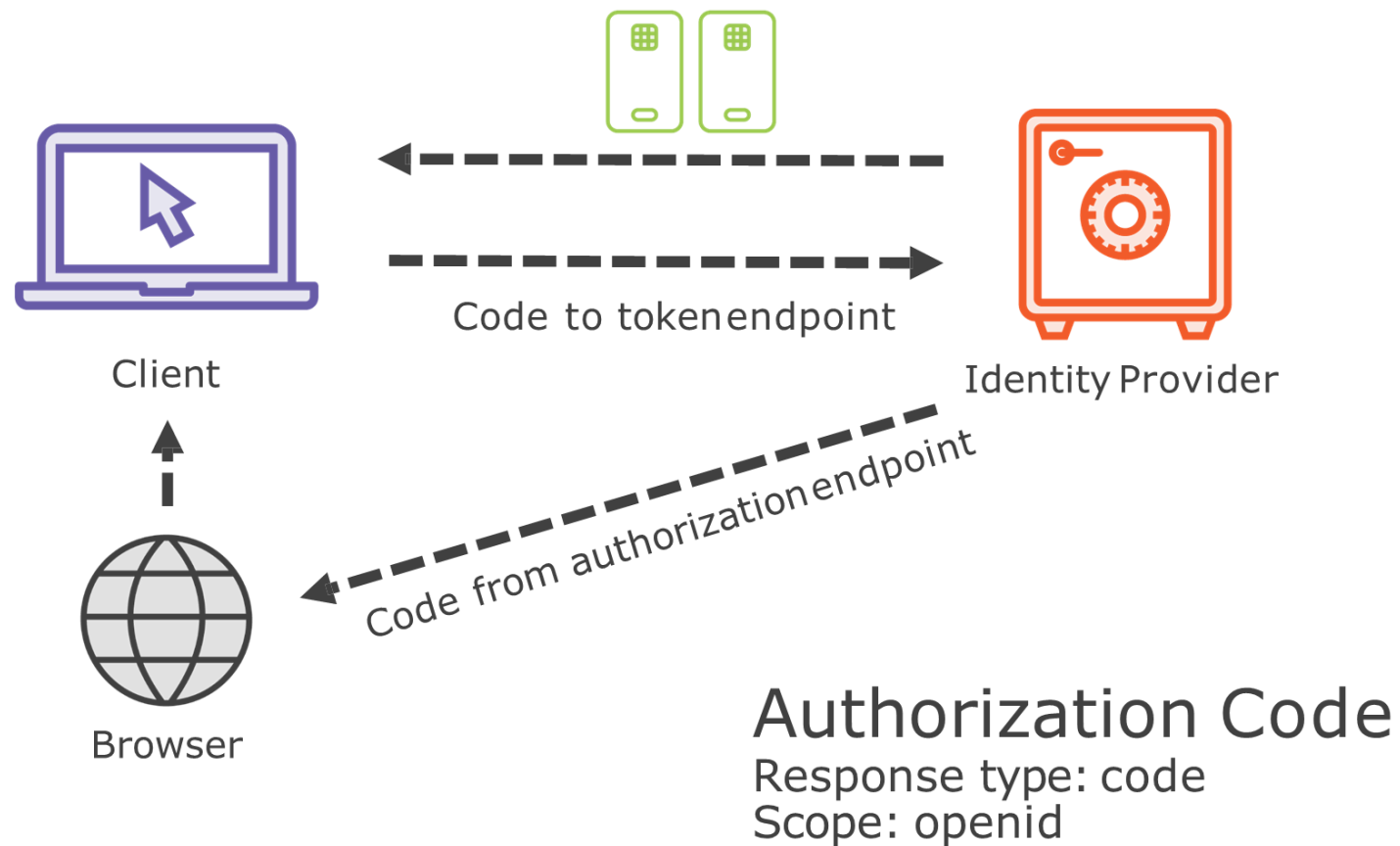
Comunicación con el Front End

Front End Inseguros

- Interacción con el punto final de autorización a través del navegador
- Redireccionamientos
- Publicación de formulario en lugar de cadena de consulta
- Front End considerado inseguro



Comunicación con el Front End





Authorization

<https://4sh.nl/PkceSpec>



Comunicación con el Front End

Single Page Applications

- “Llamadas al Backend” todavía se utiliza
- Tokens expuestos en el navegador
- El flujo del código de autorización con PKCE aún es más seguro que otras opciones
- Cliente público
- Puede desactivarse
- Las aplicaciones móviles + de escritorio también son clientes públicos



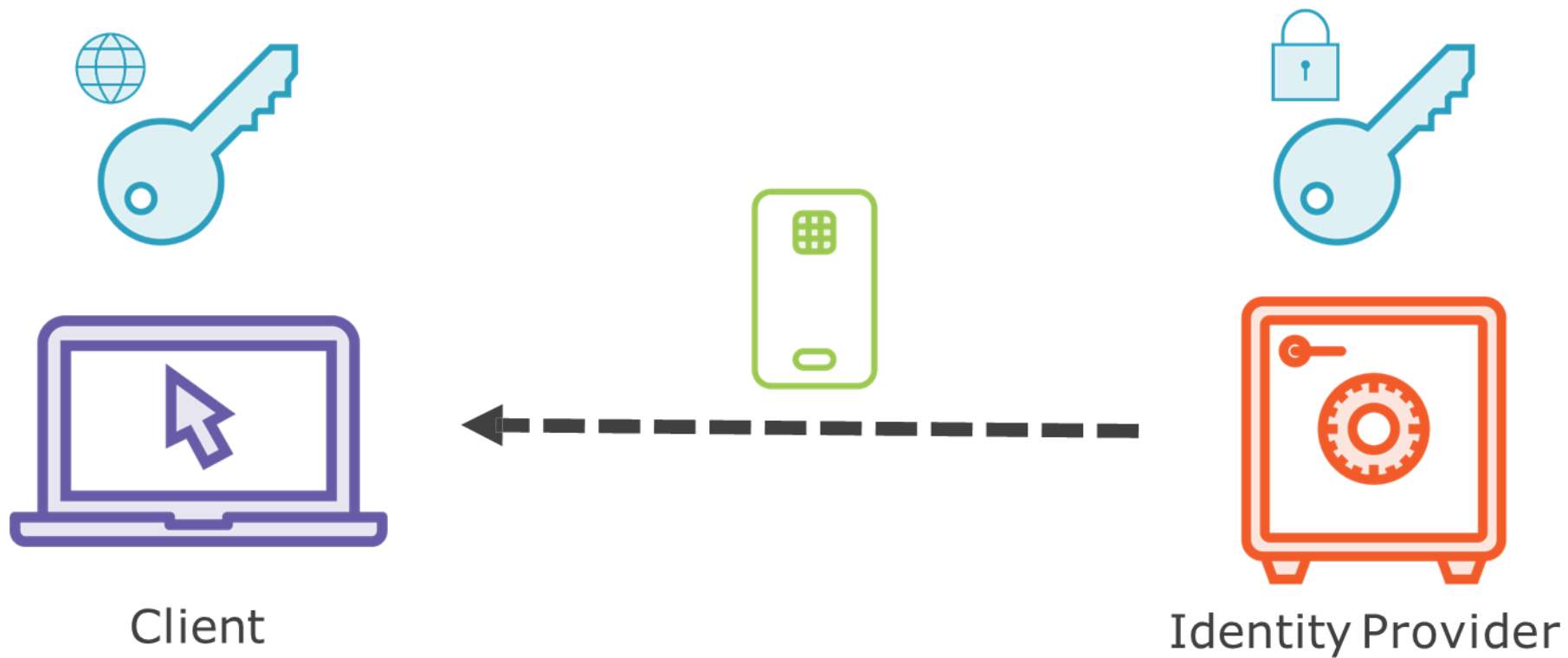
Seguridad por Token's

Verificación de tokens

- El proveedor de identidad crea un hash del contenido
- El hash se encripta con clave privada
- Adjunta el resultado (== firma) al token
- El cliente usa la clave pública para descifrar el hash
- El contenido legible es hasheado
- Compara el propio hash con el hash descifrado



Seguridad por Token's





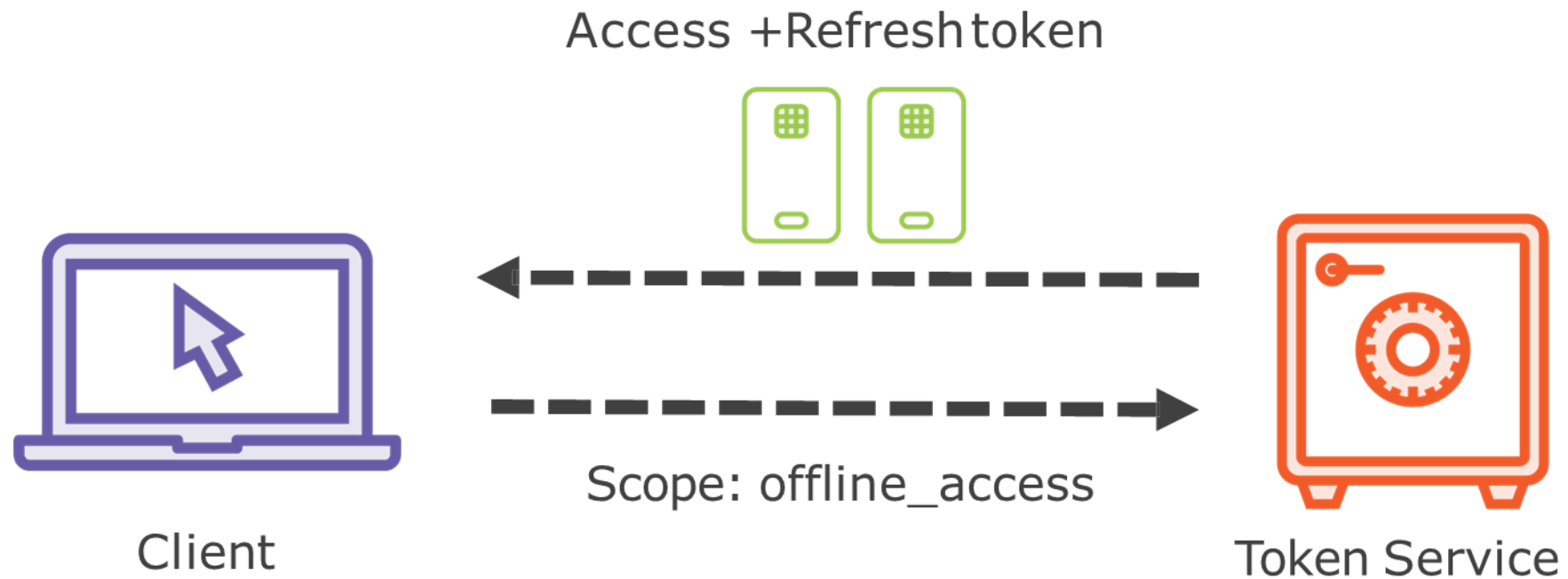
Seguridad por Token's

Refresh Tokens

- Token's separados
- Se usa para obtener un nuevo token de acceso
- El usuario no tiene que volver a autenticarse
- Mayor tiempo de vencimiento que el token de acceso

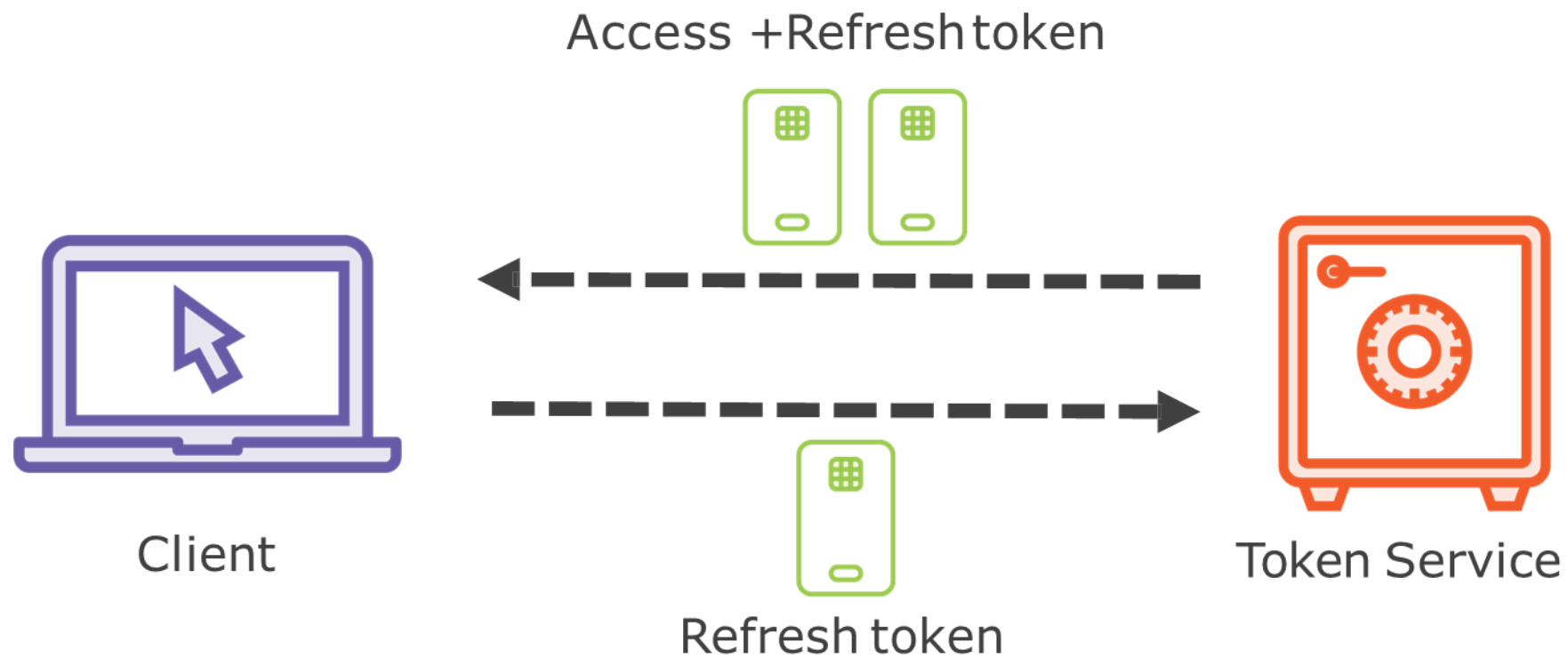


Seguridad por Token's





Seguridad por Token's





GRACIAS

POR SU PREFERENCIA