Explorando la Autenticación sin contraseña

Workshop con Transmit Security

4 Junio 2024

Equipo

Ana

Angel

Claudio

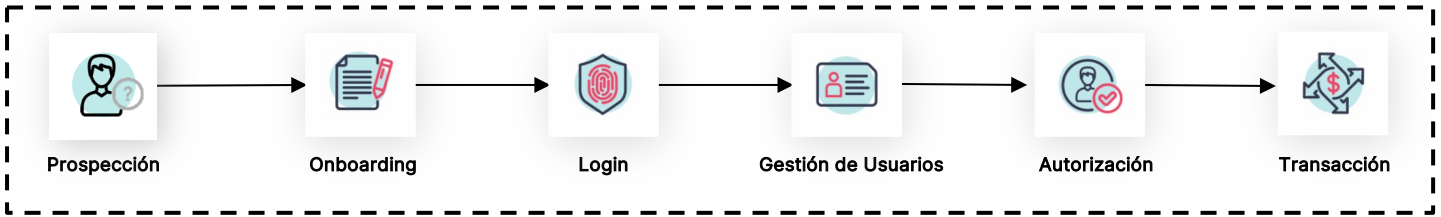
Javier

Qué vamos a hacer hoy

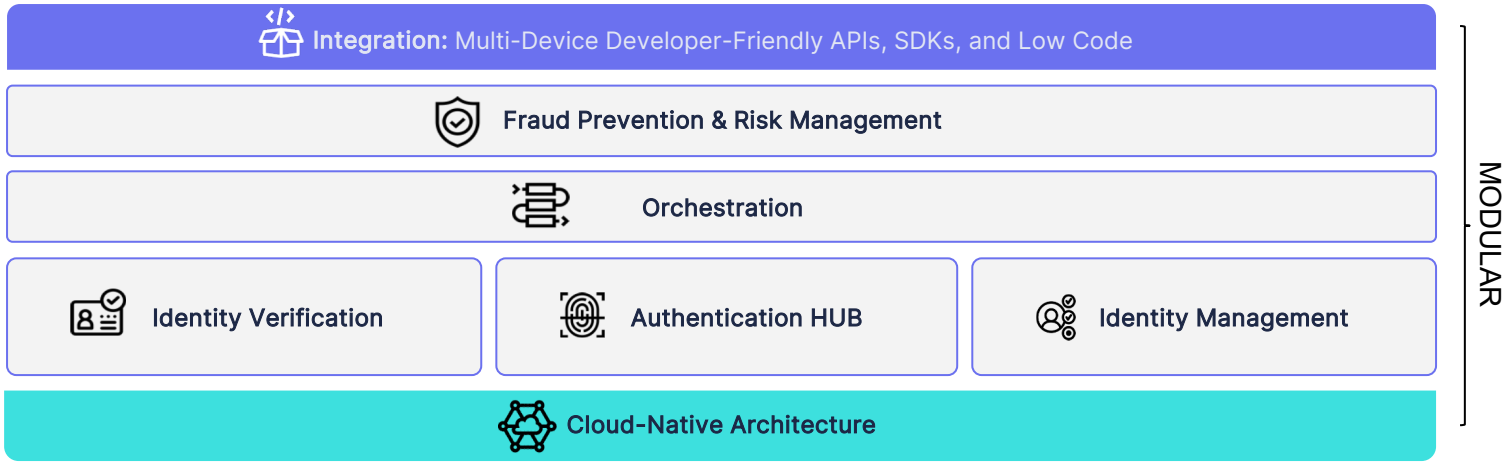
- Hablaremos de “passwordless” (FIDO2 y Passkeys)
- Integraremos el SDK de Transmit Security en una aplicación de ejemplo para pasar de autenticación basada en contraseña a autenticación “sin contraseña” (passwordless).
- Deseable para este workshop
  - Conocimiento básico de HTML, CSS, Javascript, NodeJS, Git
  - Usaremos Visual Studio Code y un navegador

Una Plataforma para gestionar la Identidad de los clientes

Pasos típicos relacionados con la identidad de cliente implementados por las aplicaciones



Una Plataforma para gestionar la Identidad de los clientes



Hub de Autenticación

Proporciona todos los métodos de autenticación en un solo servicio, incluyendo MFA resistente al phishing con Passkeys.

- Fortalece MFA con autenticación biométrica
- Ofrece un conjunto completo de métodos de inicio de sesión en un solo servicio
- Soporta todos los canales, aplicaciones y dispositivos
- Habilita la autenticación basada en riesgos, la firma de transacciones y la vinculación de dispositivos

OTP | Magic Link | Password | TOTP | FIDO2 / Passkeys | Social Login

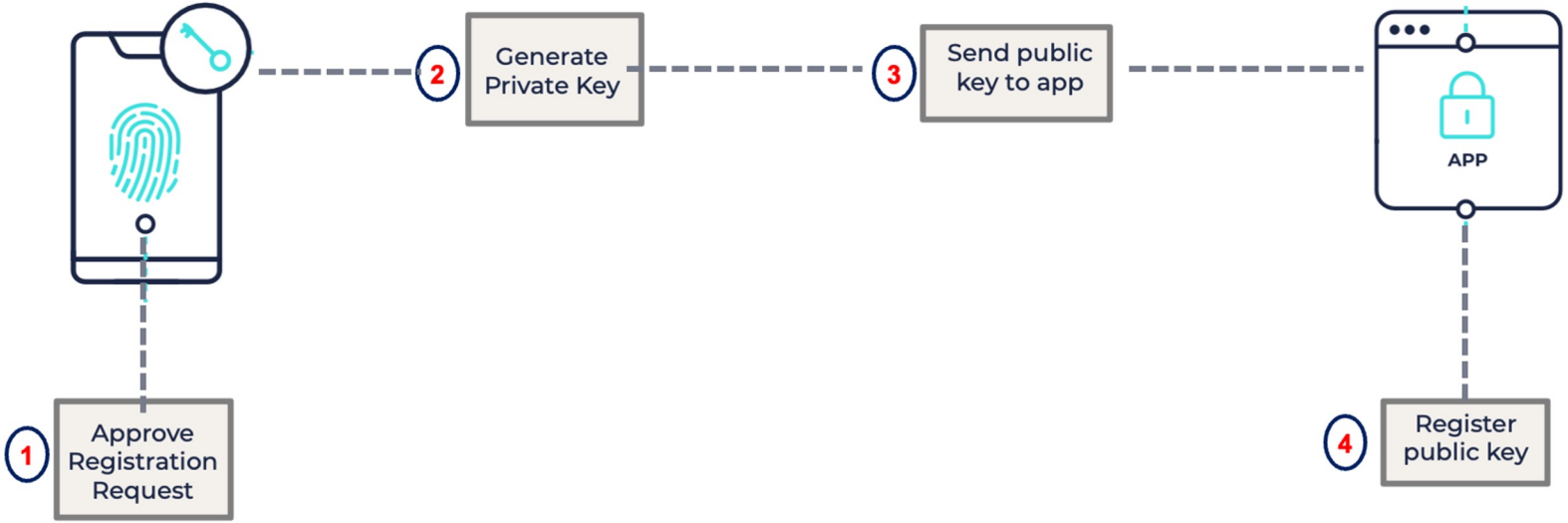
Credenciales FIDO2

- Un par de claves (2) 
- Una clave pública, registrada para el usuario
- Una clave privada. El usuario desbloquea el dispositivo para usarla

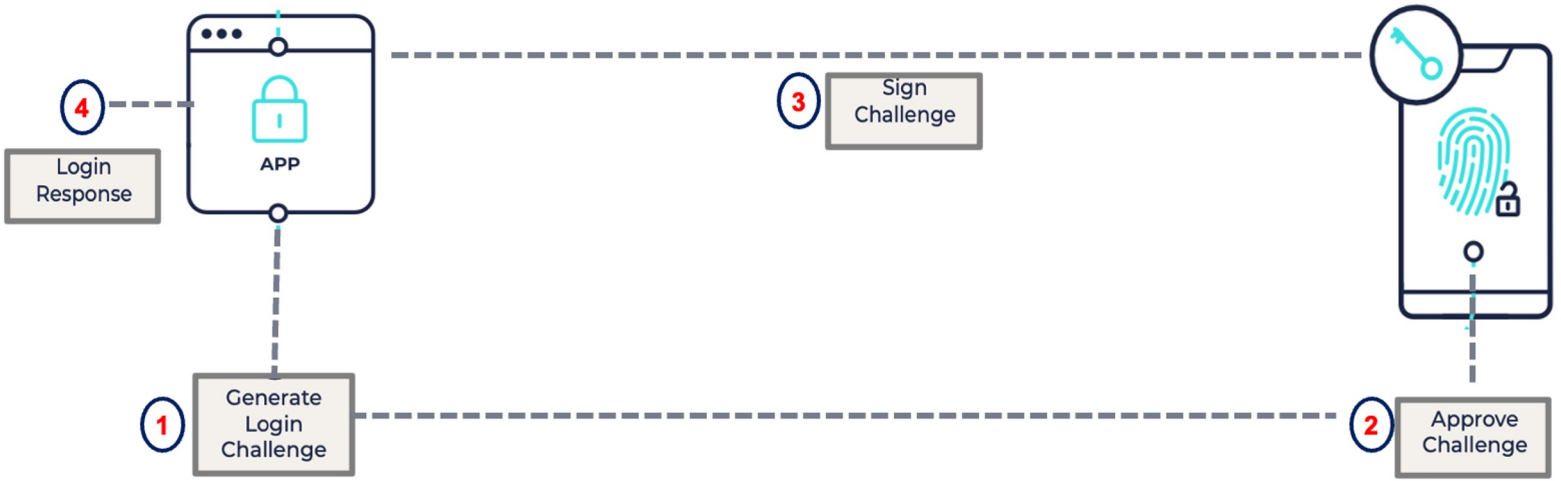
Public Key

Private Key

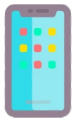
FIDO2 Flujo de Registro



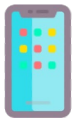
FIDO2 Flujo de Autenticación



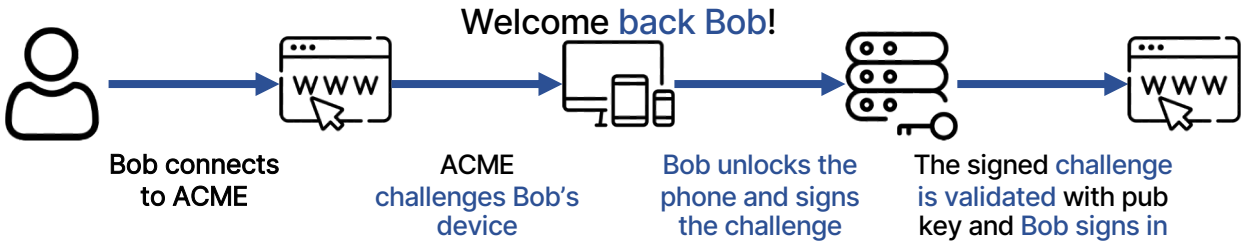
Credenciales FIDO2



Registration
(New device)



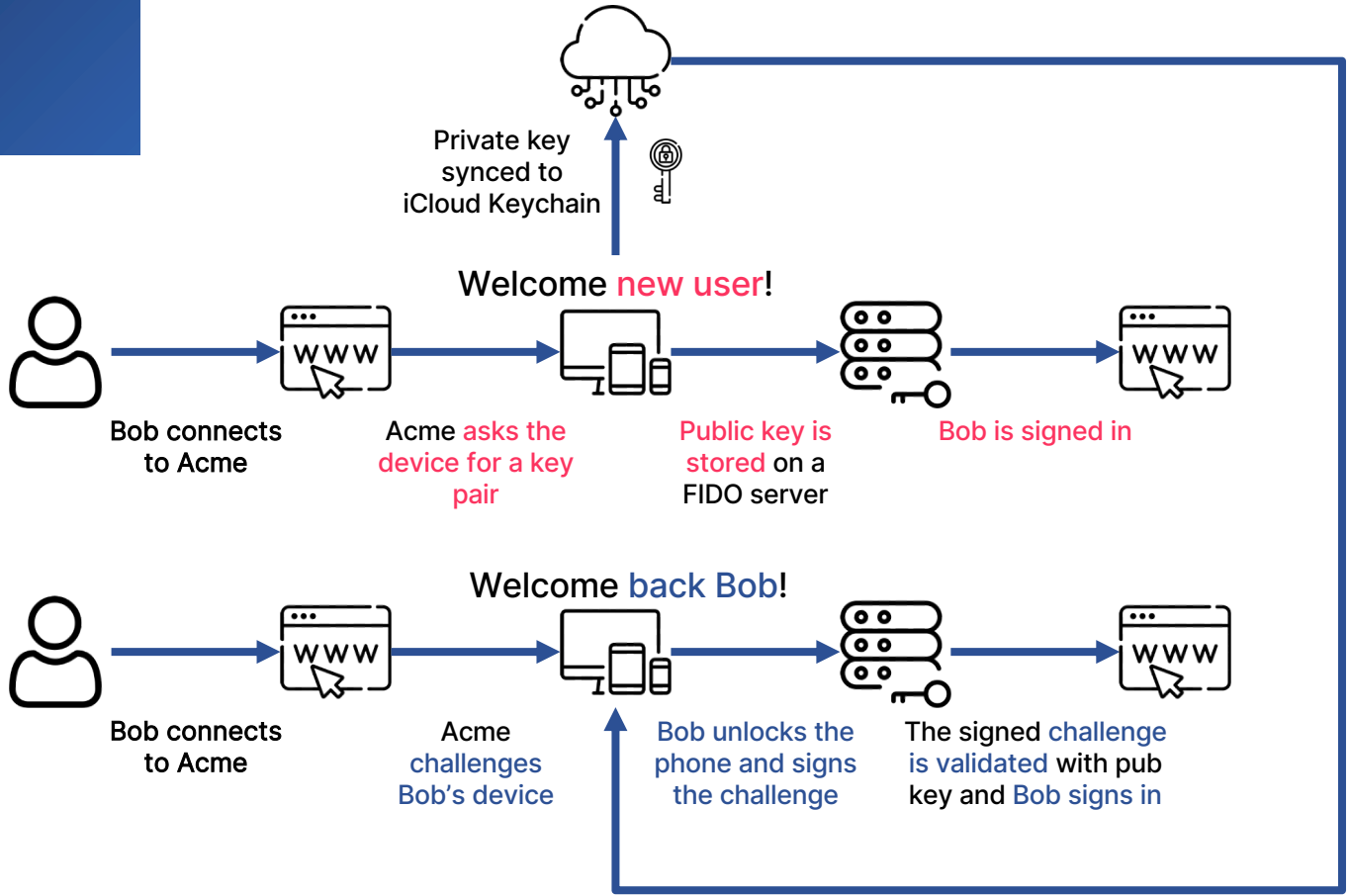
Authentication
(Known device)



Passkeys

Registration
(New device)

Authentication
(New device)



Beneficios

Seguridad

Las Passkeys (credenciales de inicio de sesión criptográficas) son únicas para cada sitio web, nunca salen del ecosistema del usuario y están protegidas por MFA. Este modelo de seguridad elimina los riesgos de phishing, todas las formas de robo de contraseñas y ataques de repetición.

Privacidad

Debido a que las Passkeys son únicas para cada sitio de internet, no pueden ser utilizadas para rastrear a los usuarios a través de diferentes sitios. Además, los datos biométricos, cuando se utilizan, nunca salen del dispositivo del usuario.

Conveniencia

Los usuarios desbloquean las credenciales de inicio de sesión criptográficas con métodos integrados simples, como lectores de huellas digitales o cámaras en sus dispositivos. Los consumidores pueden seleccionar el dispositivo que mejor se adapte a sus necesidades.

Escalabilidad

Los sitios web pueden habilitar Passkeys a través de llamadas estándar que son compatibles con los principales navegadores y plataformas en miles de millones de dispositivos que los consumidores usan todos los días.

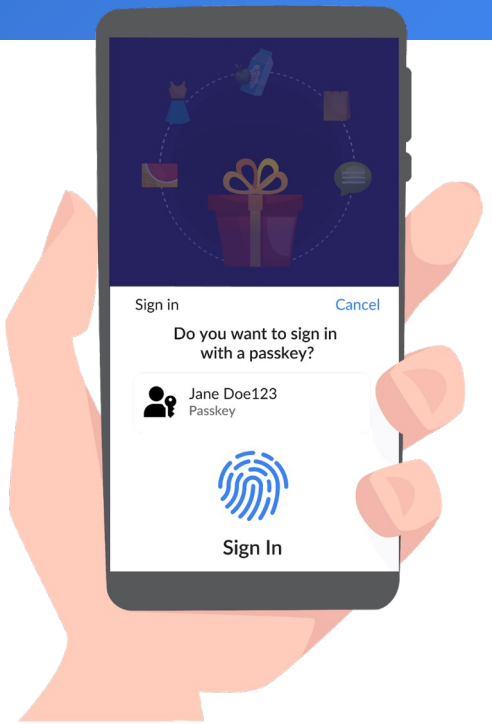
FIDO2 mejora la seguridad

Credenciales No Suplantables (*non-phisheable*):

- Elimina las Contraseñas: Utiliza criptografía de clave pública para autenticar y asegurar las comunicaciones.
- Autenticación Fuerte: Diseñado para usar formas más fuertes de autenticación, como biometría.
- Reduce los Riesgos de Brechas de Datos: Los servidores solo almacenan claves públicas, minimizando el riesgo de brechas de datos.
- Reduce las Amenazas de Phishing: Usa una clave pública única para cada sitio web, haciendo que los ataques de phishing sean ineficaces.

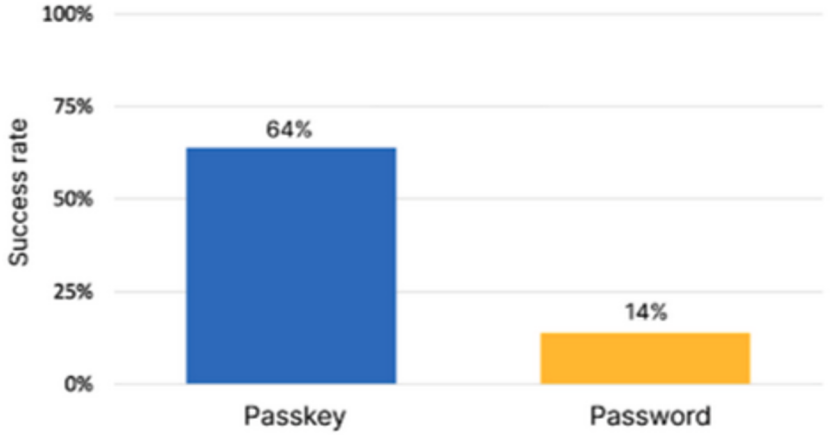
Experiencia de Usuario

- Facilidad de uso: desbloqueo del dispositivo
- Las Passkeys no se pueden “phishear”
- No hay que recordarlas
- No se pueden adivinar



Experiencia de Usuario

- Autenticarse con Passkeys es el doble de rápido que hacerlo con contraseña
- Además, el porcentaje de éxito es mayor



de Google

Transmit y Passkeys

Protege y controla el uso de credenciales sincronizadas

Recuperación de cuenta entre plataformas (Google / Apple / Microsoft)

Autenticación entre dispositivos, sin necesidad de proximidad (BLE)

Flujos de “firma de transacciones” y “aprobación”

Soporte para dispositivos no compatibles con FIDO o Passkeys

Soporte para toda la experiencia de cliente

Lo que vamos a hacer

PASO 1: Configuración de la Plataforma

- Crear Aplicación
- Configurar métodos de Autenticación
  - Passkeys
    - Relying Party ID: **localhost**
    - Relying Party Origins: **http://localhost:3001**

PASO 2: Inicializar el SDK

```
<!-- This loads the latest SDK within the major version 1. -->
<script src="https://platform-websdk.transmitsecurity.io/platform-websdk/1.x/ts-platform-websdk.js"
defer="true" id="ts-platform-script"></script>
```

```
// Initialize the SDK once it's loaded
document.getElementById('ts-platform-script').addEventListener('load', () => {
  // Configures the SDK with your client.
  tsPlatform.initialize({ clientId: [CLIENT_ID] });
});
```

PASO 3: Registro de Credenciales

```
// Check for WebAuthn Support
const isBiometricsSupported = await window.tsPlatform.webauthn.isPlatformAuthenticatorSupported();
```

```
// Registers WebAuthn credentials on the device and returns an encoded result
// that should be passed to your backend to complete the registration flow
const encodedResult = await window.tsPlatform.webauthn.register('[USERNAME]');
```

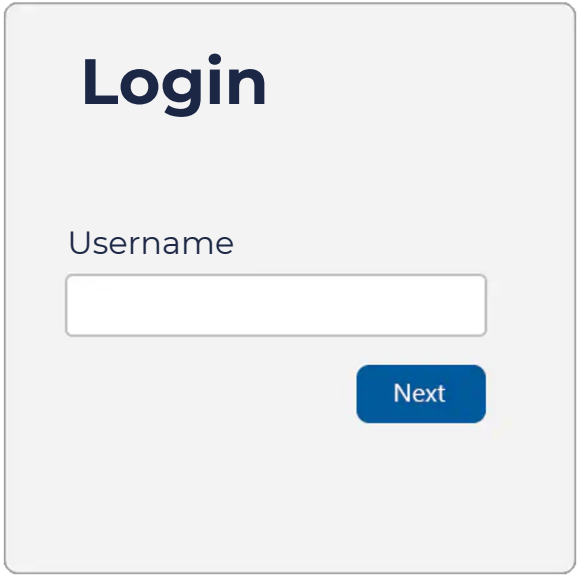
transmit security

PASO 3: Registro de Credenciales (cont.)

```
// Complete registration (back-channel)
const resp = await fetch(`https://api.transmitsecurity.io/cis/v1/auth/webauthn/external/register`, {
  method: 'POST',
  headers: {
    'Content-Type': 'application/json',
    Authorization: 'Bearer [TOKEN]'// Client access token
  },
  body: JSON.stringify({
    webauthn_encoded_result: '[ENCODED_RESULT]', // Returned by register() SDK call
    external_user_id: '[EXTERNAL_USER_ID]'// Identifier of the user in your system
  })
});

const data = await resp.json();
console.log(data);
```

PASO 4: Autenticación



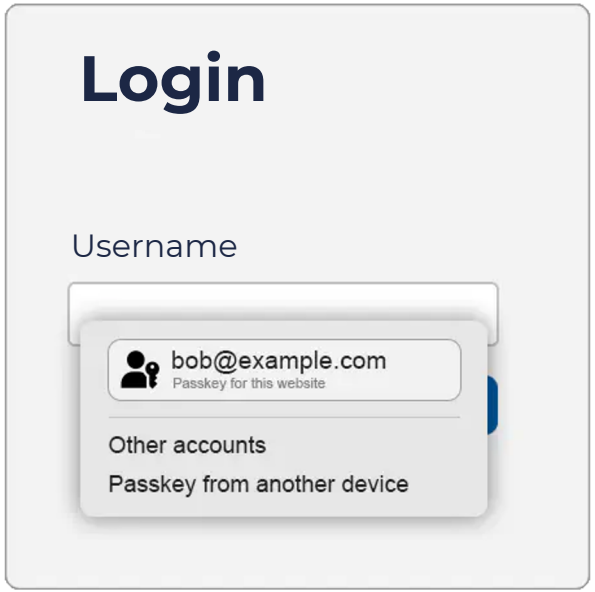
PASO 4: Autenticación

```
// Authenticates WebAuthn credentials on the device (username is optional)
const webauthnEncodedResult = await window.tsPlatform.webauthn.authenticate.modal("USERNAME");
```

```
const resp = await fetch(`https://api.transmitsecurity.io/cis/v1/auth/webauthn/authenticate`, {
  method: 'POST',
  headers: {
    'Content-Type': 'application/json',
    Authorization: 'Bearer [TOKEN]' // Client access token
  },
  body: JSON.stringify({
    webauthn_encoded_result: '[ENCODED_RESULT]' // Returned by authenticate.modal() SDK call
  })
});

const data = await resp.json();
console.log(data);
```

PASO 4: Autenticación (cont.) – “Autofill”



PASO 4: Autenticación (cont.) – “Autofill”

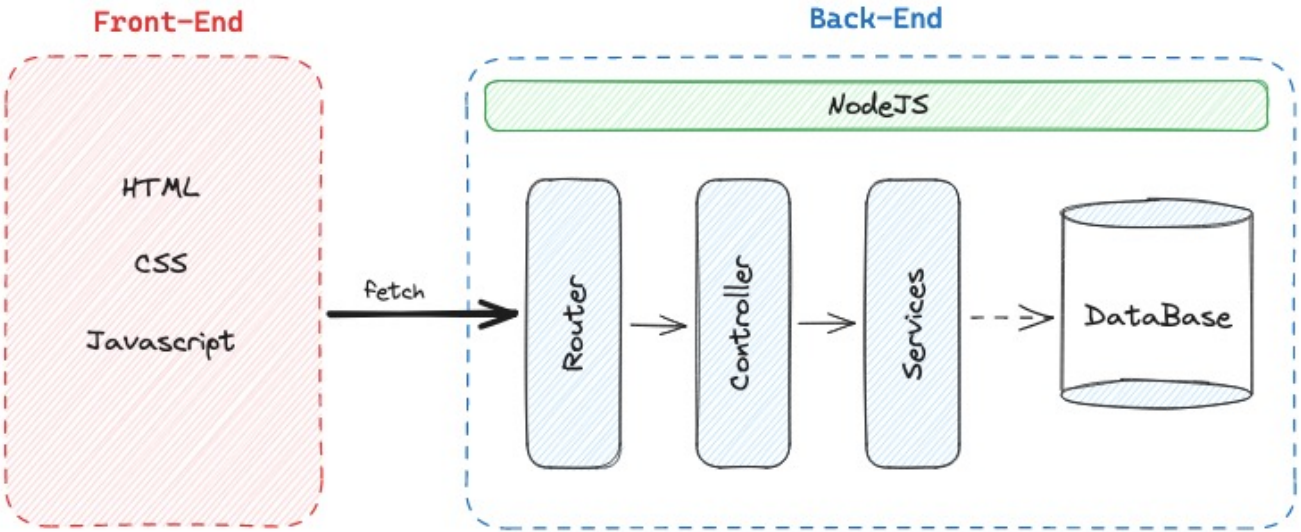
```
// Verify if autofill is supported
const isAutofillSupported = await window.tsPlatform.webauthn.isAutofillSupported();
```

```
// Activate the passkey autofill
window.tsPlatform.webauthn.authenticate.autofill.activate({
  onSuccess: handleSuccessfulPasskeyValidation, // Handle successful authentication
  onError: handleAutofillError, // Handle error or passkey cancellation
});

async function handleSuccessfulPasskeyValidation({webauthnEncodedResult}) {
   // Send the encoded result to your backend to complete the authentication flow
}

async function handleAutofillError(error) {
  if (error.errorCode === 'autofill_authentication_aborted') return; // Authn canceled by user
  console.log(error);
}
```

Arquitectura



Conclusión

- Las Passkeys como mecanismo de autenticación ofrecen beneficios:

  ↑ Seguridad ↑ UX ↑ Privacidad ↑ Escalabilidad

- Transmit Security ofrece una forma sencilla y rápida para empezar a utilizar autenticación sin contraseña basada en Passkeys (y/o varios otros mecanismos de autenticación)

GRACIAS

transmit security