Integrando Anti-Fraude en tu aplicación

Workshop con Transmit Security

10 Septiembre 2024

Equipo

Ana

Ángel

Claudio

Javier

Qué vamos a hacer hoy

- Hablaremos del estado del arte de las herramientas Anti-Fraude.

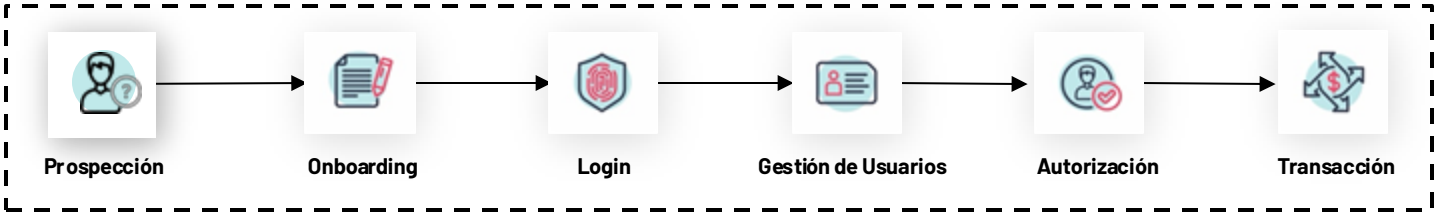
- Integraremos el servicio **Detection and Response** de la plataforma **Mosaic by Transmit Security** en una aplicación de ejemplo para recoger telemetría, detectar riesgo y realizar acciones en consecuencia.

- Deseable para este workshop
  - Conocimiento básico de HTML, CSS, Javascript, NodeJS, Git
  - Usaremos Visual Studio Code y un navegador

Mosaic by Transmit Security:

Plataforma para gestionar la Identidad de los clientes

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



Transmit Security: Innovando por más de 10 años

mosaic
BY TRANSMIT SECURITY

Orquestación:
Integraciones simplificadas, políticas y toma de decisiones, flujos de trabajo de los journeys de cliente

Gestión de Identidades:
Perfiles de usuario, autorización, SSO

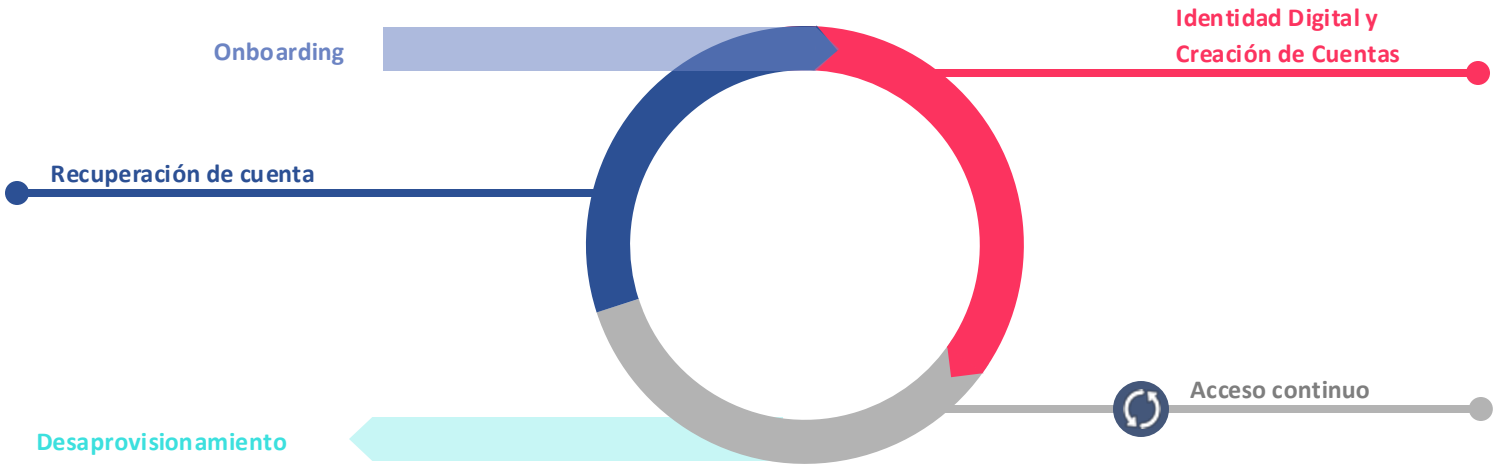
Autenticación:
Passkeys, MFA sin contraseña, enlaces mágicos, etc.

Verificación de Identidad:
Verificación de documentos y bases de datos, prueba de vida y selfis, inteligencia contra el fraude integrada

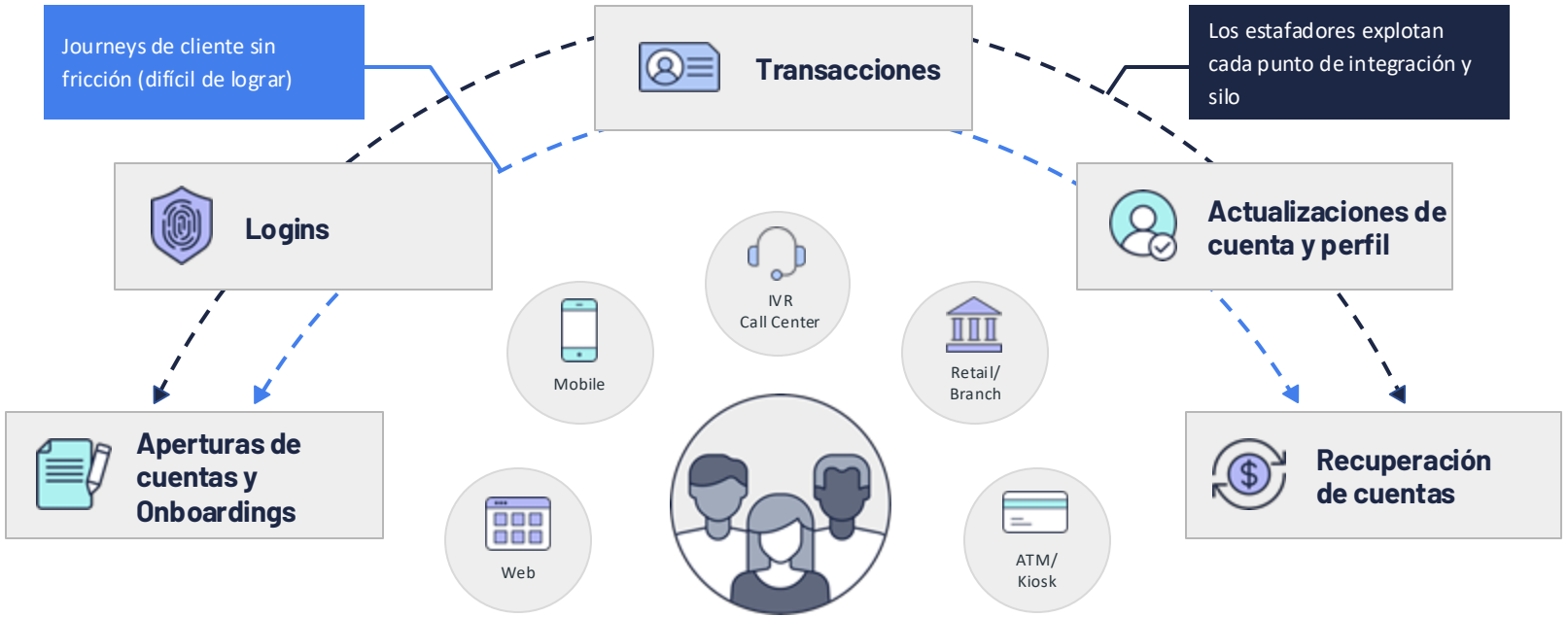
Detection and Response:
Detección de fraude multi-modo, en tiempo real y post-detección basada en IA automatizada.

transmit security

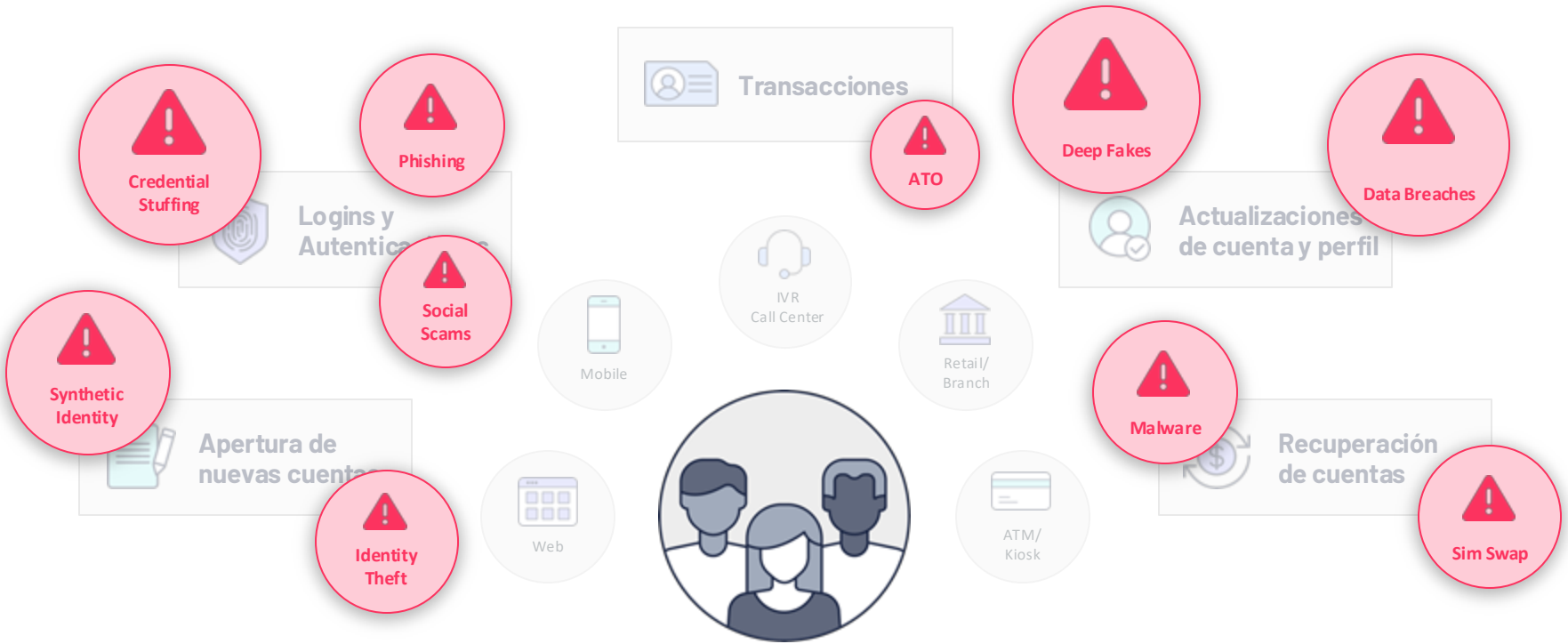
Ciclo de Vida de la Identidad



Es difícil balancear CX/UX y Seguridad



Panorama de Amenazas



Enfoque Tradicional al Fraude

Múltiples Proveedores
Soluciones puntuales para casos de uso específicos: bots, dispositivos, comportamiento, red, etc.

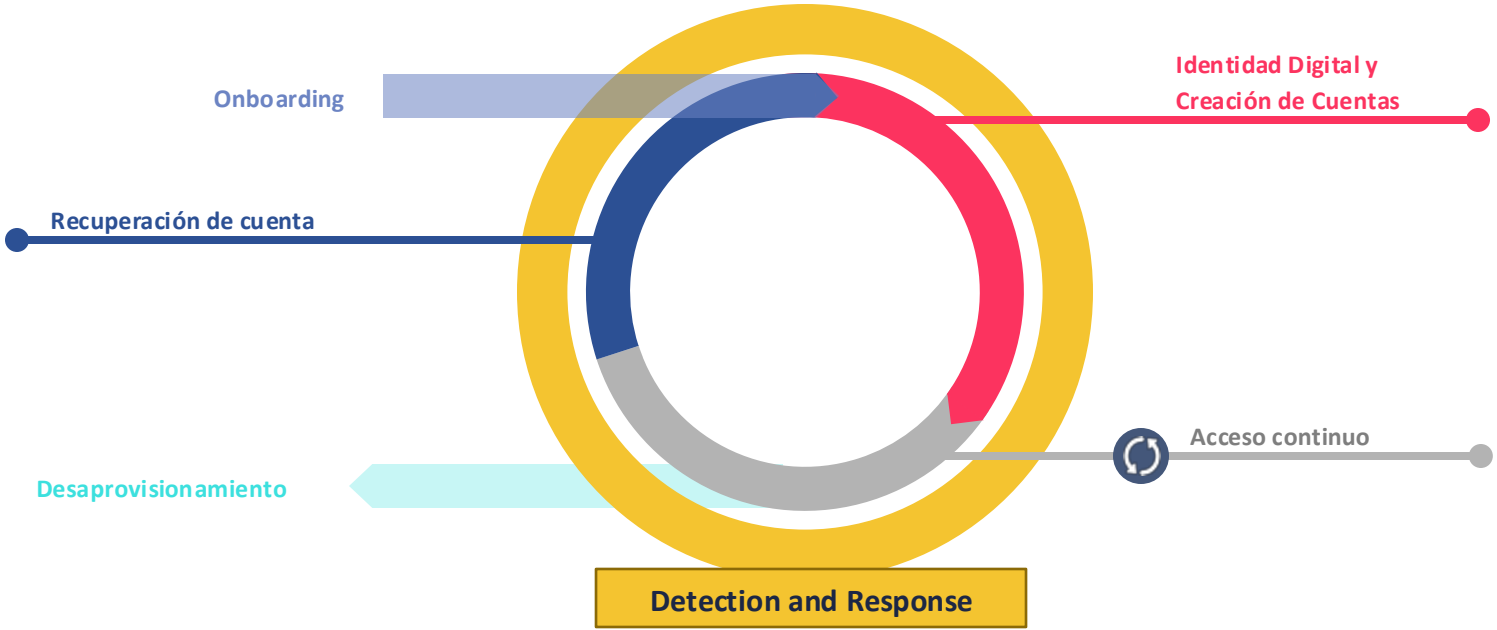
Múltiples Mecanismos de Puntuación
Mecanismos de puntuación aislados que no se comunican entre sí

Orquestación de "Cosecha Propia"
Lagos de Datos (Data lakes), integraciones complejas, normalización de datos, alto coste operativo

Basado en Reglas
Creación y mantenimiento de cientos o miles de reglas

¿Qué hace único a Detection and Response?

Asegurando el Ciclo de Vida de la Identidad



Principios Clave de Detection and Response

Detección de Riesgos basada en IA y ML a lo largo del Ciclo de Vida
Ofrece decisiones en tiempo real basadas en la identidad en cualquier punto de interacción del cliente

La Adaptabilidad del Modelo es Crítica
Los métodos de fraude están evolucionando al ritmo de la IA Generativa; los modelos de detección deberían hacerlo de igual modo.

Automatización
Automatización del ML, investigación de casos y flujos de trabajo integrados que optimizan tus operaciones antifraude

Calidad de Datos e IA Confiable, dirigida por Expertos en Fraude
Cientos de señales de telemetría y correlaciones, procesadas con un enfoque de AI-in-the-loop

Qué NO es Detection and Response

Basado en Reglas
Despídete de configurar y mantener miles de complicadas reglas

Experimentar con Datos Dispares
DRS es una única base de código que orquesta cientos de señales para un enfoque integral en la detección de fraude

Caja Negra
Piensa en una "caja de cristal" con total transparencia en tus modelos y en las razones detrás de las recomendaciones

Dependencia de Servicios Profesionales
DRS es 100% autogestionado. Deja de gastar en declaraciones de trabajo de SSPP

Gestión del Fraude con Detection and Response

DETECTA: Obtén recomendaciones unificadas y transparentes en tiempo real para cada solicitud mediante la detección de anomalías basada en IA, que sintetiza inteligencia de múltiples métodos de detección.

RESPONDE: Utiliza estas recomendaciones para reducir automáticamente la fricción para usuarios confiables, mientras bloqueas o desafías solicitudes sospechosas.

ANALIZA: Aprovecha la IA generativa, análisis offline y paneles centralizados para acelerar el análisis de datos y visualizar rápidamente campañas a gran escala, anillos de fraude, cuentas mule y tendencias de ataque.

INVESTIGA: Accede a todas las solicitudes, señales de riesgo, autenticadores y otros datos de identidad de un usuario en un solo lugar, y utiliza herramientas de gestión de casos y un único panel para simplificar y agilizar las investigaciones.

ADAPTA: Ajusta y adapta los mecanismos de seguridad usando herramientas de bajo código y flujos de trabajo automatizados para prepararte para futuros ataques. Los modelos de detección se mejoran y actualizan continuamente con feedback automatizado, análisis de anomalías y estudios internos de amenazas.

Ejercicio práctico: Qué vamos a hacer

PASO 1: Configuración de la Plataforma

- Usaremos la misma Aplicación que en el workshop anterior

- Habíamos configurado los métodos de Autenticación
  - Passkeys
    - Relying Party ID: **localhost**
    - Relying Party Origins: **http://localhost:3001**

PASO 2: Inicializar el SDK (ya lo hicimos)

```
<!-- 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>
```

```
await window.tsPlatform.initialize({
  clientId: import.meta.env.VITE_TS_CLIENT_ID,
  webauthn: { serverPath: import.meta.env.VITE_TS_BASE_URL },
});
```

PASO 3: Chequear el riesgo/confianza en el login

Client Side

```
// Trigger action event
const actionResponse = await window.tsPlatform.drs.triggerActionEvent(RISK_ACTIONS.LOGIN);
```

PASO 3: Chequear el riesgo/confianza en el login (cont.)

Server Side

```
/**
 * Fetch DRS recommendation
 * @param {String} actionToken Obtained from the SDK
 * @param {String} riskClientAccessToken  ClientAccessToken for the DRS API
 */
export const getDRSRecommendation = async (actionToken, riskClientAccessToken) => {
  try {
    const query = new URLSearchParams({ action_token: actionToken }).toString();

    const resp = await fetch(`${process.env.VITE_TS_BASE_URL}/risk/v1/recommendation?${query}`, {
      method: 'GET',
      headers: {
        Authorization: `Bearer ${riskClientAccessToken}`,
      },
    });

    const data = await resp.json();
    //console.log(`DRS Recommendation: ${JSON.stringify(data, null, 2)}`);
    return data;
  } catch (error) {
    console.error(`${ERROR_RISK_GET_RECOMMENDATION}: ${error.message}`);
    throw new Error(ERROR_RISK_GET_RECOMMENDATION);
  }
};
```

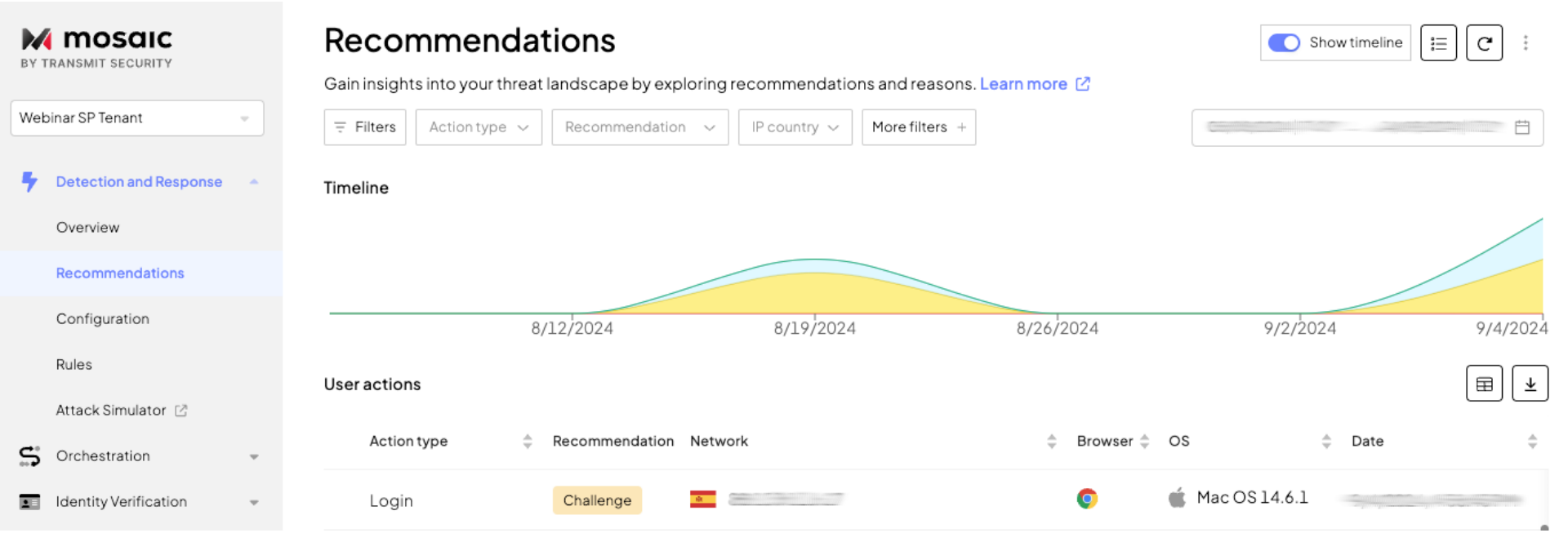
PASO 3: Chequear el riesgo/confianza en el login (cont.)

Server Side

```
/**
 * Manages risk for login
 * @param {String} actionToken DRS action token
 * @returns recommendation
 * @throws Error based on the recommendation (only for DENY)
 */
const manageRiskLogin = async (actionToken) => {
  try {
    const recommendation = await getRiskRecommendation(actionToken);
    console.log(recommendation); // TODO: for visibility only, remove in production

    // Login Business Logic:
    //   Only deny login if recommendation is DENY, otherwise continue
    if (recommendation.recommendation.type === RECOMMENDATIONS.DENY) {
      throw new Error(ERROR_RISK_DENY);
    }
    return recommendation;
  } catch (error) {
    console.error(error.message);
    throw new Error(error.message);
  }
};
```

PASO 3: Chequear el riesgo/confianza en el login (cont.)



PASO 4: Refactoring

Detection and Response:
Detección de fraude multi-modo, en tiempo real y post-detección basada en IA automatizada.

- Login
- Registration
- Checkout

PASO 5: Añadir información en el checkout

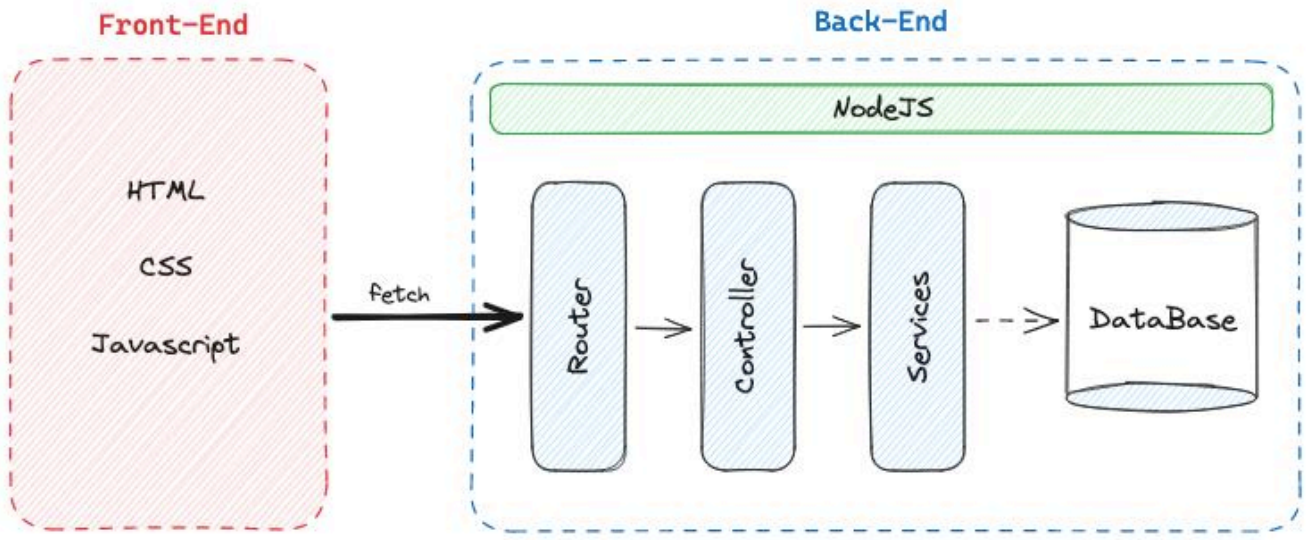
```
// Trigger action event: checkout
const actionResponse = await window.tsPlatform.drs.triggerActionEvent(RISK_ACTIONS.CHECKOUT, {
  transactionData: {
    amount: totalAmount,
    currency: 'USD',
    reason: `Purchased ${totalNFTs} NFTs for ${totalAmount} USD.`,
    transactionDate: Date.now(),
    payer: {
      name: user.userid,
    },
  },
});
```

PASO 6: Reportar el resultado de la acción

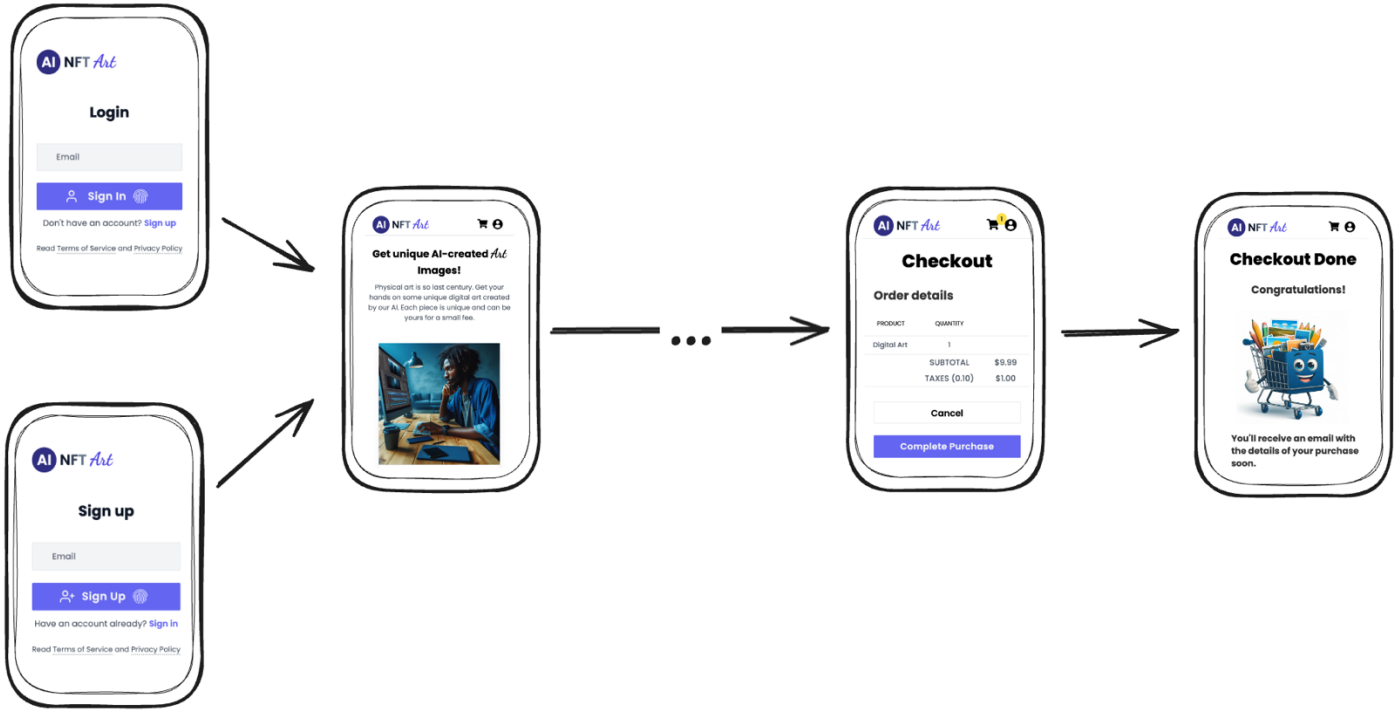
```
// Report action result
try {
  const body = {
    action_token: actionToken,
    result: result,
  };
  if (userId) body.user_id = userId;
  if (challengeType) body.challenge_type = challengeType;

  const resp = await fetch(`${process.env.VITE_TS_BASE_URL}/risk/v1/action/result`, {
    method: 'POST',
    headers: {
      'Content-Type': 'application/json',
      Authorization: `Bearer ${riskClientAccessToken}`,
    },
    body: JSON.stringify(body),
  });
  ...
```

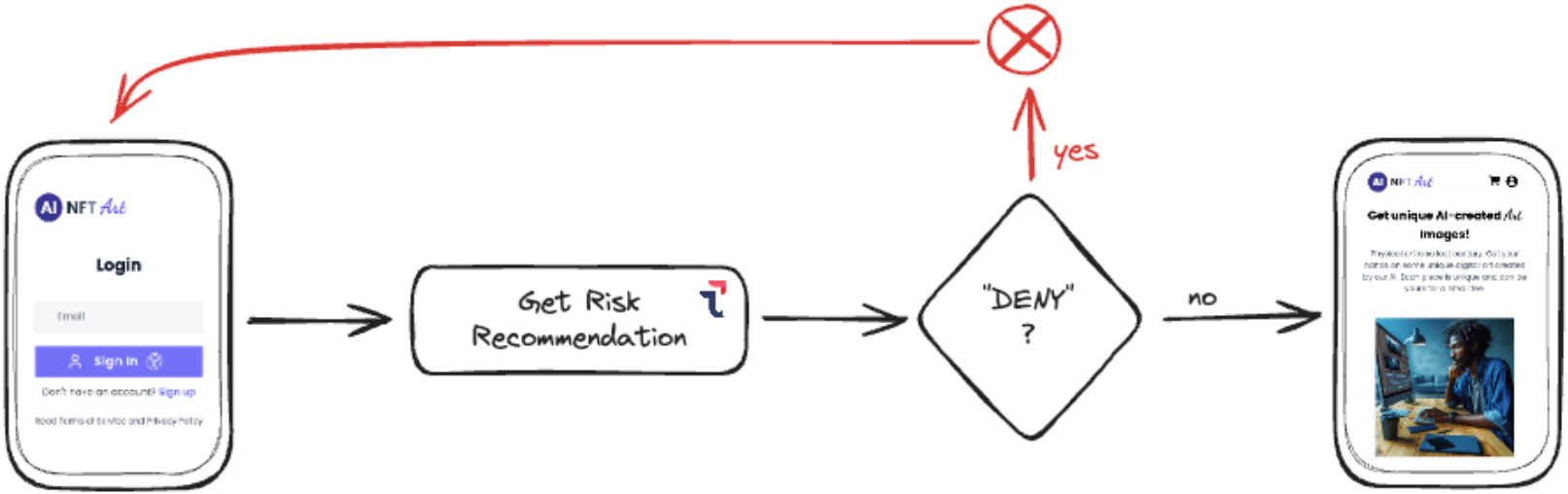
Arquitectura



Nuestra Aplicación

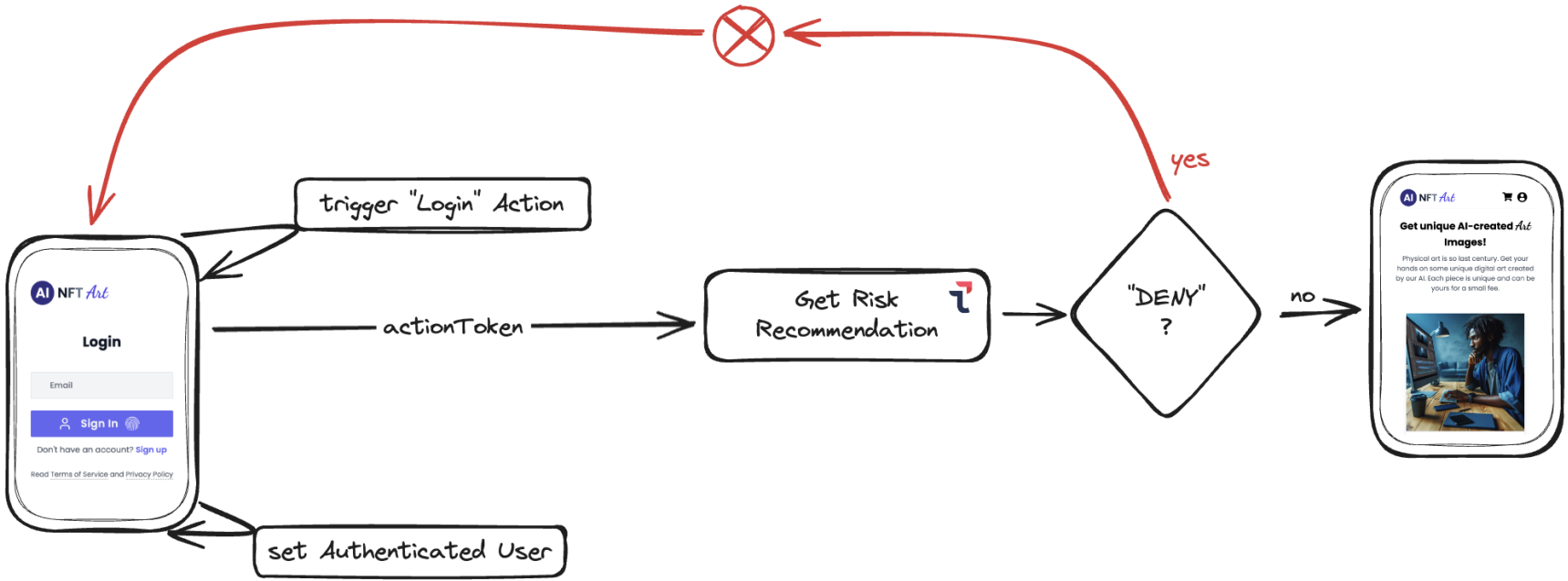


Nuestra Aplicación con Gestión del Fraude (Paso 1)

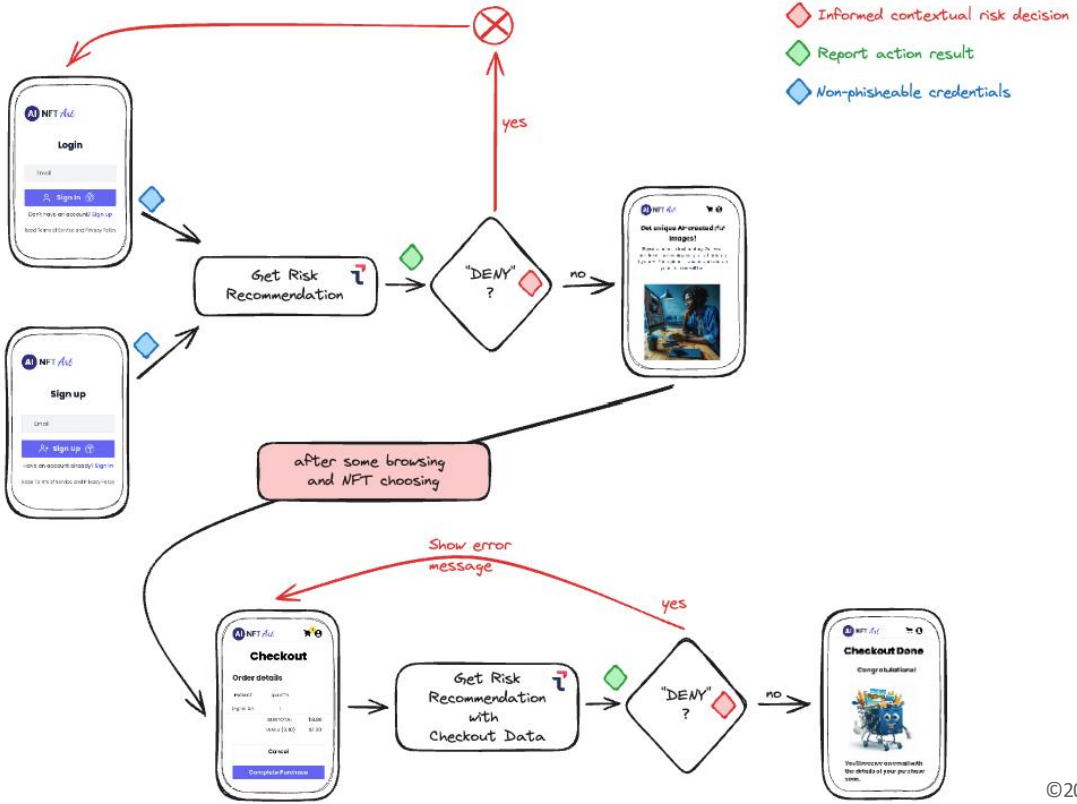


https://github.com/TransmitSecurity/workshop-latam → doc → 02 Instructions.md
https://github.com/TransmitSecurity/workshop-latam/blob/main/doc/02%20Instructions.md

Nuestra Aplicación con Gestión del Fraude (Paso 1)



Nuestra Aplicación con Gestión del Fraude (Paso 2)



Conclusión: La Evolución de la Seguridad de la Identidad

Una solución de fraude efectiva debe poder operar a lo largo de todo el journey del cliente, en cada punto de contacto, no solo en el login o en el backend después de que se hayan iniciado las transacciones o las compras.

Además, una solución efectiva debe tener un enfoque integrado para todas las técnicas comunes de los defraudadores: bots, secuestro de cuentas y malware.

mosaic
BY TRANSMIT SECURITY

Transmit Security ofrece una solución de gestión de fraude moderna, sencilla y rápida, basada en ML e IA, enfocada en mantener la más alta seguridad con la mejor experiencia de usuario.

↑ Seguridad ↑ UX ↑ Privacidad ↑ Escalabilidad

transmit security

GRACIAS

Ana de Jorge — ana.dejorge@transmitsecurity.com
Ángel Nogueras — angel.nogueras@transmitsecurity.com
Claudio Silotto — claudio.silotto@transmitsecurity.com
Javier Jarava — javier@transmitsecurity.com