18 DE JULIO DE 2025

PLATAFORMA DE AUTENTICACIÓN DE LA IDENTIDAD DIGITAL ID-PERÚ

ESPECIFICACIONES TÉCNICAS

Descripción:	Este documento proporciona informacion para la implementación del servicio ID-PERÚ.
Versión:	v 2.2
Autor:	Equipo de Servicios Digitales – RENIEC
Estado:	Aprobado

EQUIPO DE SERVICIOS DIGITALES
SUB DIRECCIÓN DE SERVICIOS DE CERTIFICACIÓN DIGITAL
DIRECCIÓN CERTIFICACIÓN Y SERVICIOS DIGITALES

CESAR ROBERTO ROSALES MAQUERA
Sub Director de Servicios de Certificación Digital
REGISTRO NACIONAL DE IDENTIFICACION Y ESTADO CIVIL

Firmado digitalmente por: ROSALES MAQUERA Cesar Roberto FAU 20295613620 soft
Motivo: Soy el autor del documento
Fecha: 31/07/2025 17:52:12-0500

1. Contenido

1. INTRODUCCIÓN

La Plataforma de Autenticación de la Identidad Digital ID Perú es un servicio desarrollado para habilitar mecanismos seguros de verificación de identidad digital, dirigidos a entidades públicas y privadas dentro del territorio nacional. Su propósito es validar de forma electrónica y confiable la identidad de los ciudadanos peruanos en los distintos servicios digitales que estas entidades ofrecen.

Esta plataforma, administrada por el Registro Nacional de Identificación y Estado Civil (RENIEC), estandariza y fortalece el proceso de autenticación digital, integrándose de forma transparente con las aplicaciones del sector público y privado.

ID Perú prioriza características críticas para los sistemas modernos, tales como la seguridad criptográfica, alta disponibilidad, confiabilidad operativa y rendimiento optimizado. Para ello, implementa tecnologías robustas como certificados digitales X.509, protocolos de autenticación federada (OpenID Connect) y factores biométricos (facial, dactilar, etc.), permitiendo validar la identidad de los ciudadanos con integridad, trazabilidad y protección contra alteraciones.

Gracias a esta arquitectura, las organizaciones pueden ofrecer a sus usuarios finales una experiencia de autenticación uniforme, segura y eficiente, cumpliendo con estándares nacionales e internacionales en materia de identidad digital.

2. OBJETIVOS

- Validar electrónicamente la identidad de los ciudadanos peruanos, permitiendo su acceso seguro a servicios digitales ofrecidos por entidades públicas y privadas, mediante mecanismos confiables y estandarizados.
- Unificar y simplificar el proceso de autenticación, proporcionando una única interfaz de acceso para múltiples plataformas del sector público y privado, compatible con distintos factores de autenticación (biometría, certificados digitales, entre otros).
- Garantizar altos niveles de seguridad, integridad y disponibilidad, asegurando que cada transacción de autenticación sea resistente a fraudes, trazable y ejecutada con el mejor rendimiento posible, conforme a buenas prácticas de ciberseguridad y diseño de sistemas distribuidos

3. PRINCIPIOS Y CONSIDERACIONES DE DISEÑO

La Plataforma de Autenticación ID Perú ha sido desarrollada bajo una arquitectura modular y escalable, pensada para facilitar la integración,

mantenimiento y evolución del servicio en entornos heterogéneos. Entre sus principales fundamentos de diseño se destacan:

- ✓ Modularidad y extensibilidad: La plataforma está estructurada en componentes desacoplados que permiten su adaptación a nuevos requerimientos tecnológicos sin afectar la estabilidad del sistema.
- ✓ Cumplimiento de estándares abiertos: Se implementa el protocolo OpenID Connect (OIDC), lo que asegura compatibilidad con múltiples tecnologías y facilita la interoperabilidad con sistemas de terceros.
- ✓ Facilidad de integración y usabilidad: Se proporciona una interfaz de autenticación unificada, intuitiva y acompañada de documentación clara, orientada a equipos de desarrollo.
- ✓ Seguridad de extremo a extremo: La autenticación está respaldada por el estándar OpenID Connect Core, asegurando el intercambio seguro de tokens y la validación criptográfica de identidades.
- ✓ Flexibilidad en los métodos de autenticación: Se soportan múltiples flujos de autenticación (biometría, certificados digitales, OTP, entre otros), permitiendo adaptar el proceso según el nivel de seguridad requerido.
- ✓ Optimización del rendimiento: El flujo de autenticación está diseñado para ejecutarse con mínima latencia, permitiendo respuestas rápidas y una experiencia fluida para el usuario final.

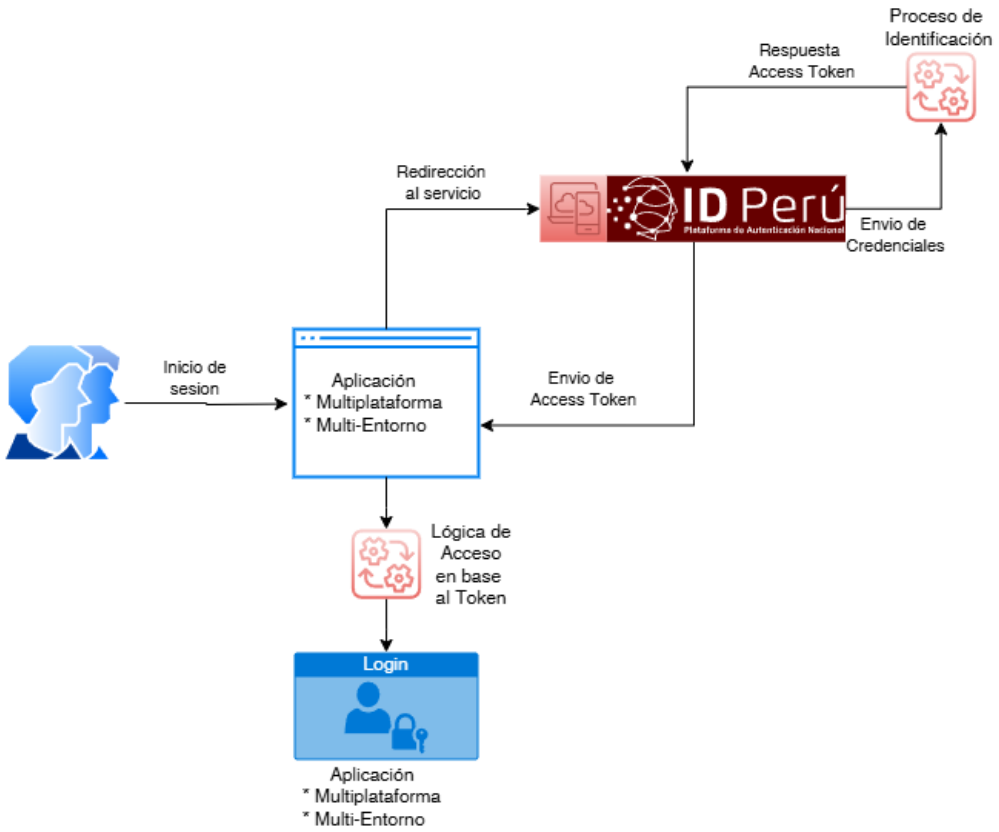

Imagen 1: Proceso de identificación

4. AUTORIZACIÓN PARA ACCEDER AL SERVICIO

Para integrar y consumir los servicios de autenticación digital ofrecidos por la plataforma ID Perú, las entidades —tanto del sector público como privado— deben cumplir con los siguientes pasos administrativos y técnicos:

1. Suscripción de Convenio: Como primer requisito, la entidad interesada debe firmar un convenio interinstitucional con RENIEC, que habilite legalmente el acceso al servicio ID Perú bajo los términos definidos. Cualquier intento de acceso no autorizado o sospechoso será evaluado por RENIEC, a fin de tomar las medidas correspondientes. Asimismo, RENIEC se reserva el derecho de resguardar esta información conforme a sus políticas de seguridad y protección de datos.
2. Solicitud de Credenciales Técnicas: Una vez suscrito el convenio, la entidad debe solicitar sus identificadores técnicos.
3. Evaluación y Emisión de Identificadores: RENIEC evaluará la solicitud enviada y, en caso de ser aprobada, procederá con la generación y entrega de las credenciales necesarias para la integración:
    - Client ID: Identificador único de la entidad integradora.
    - Client Secret: Clave secreta asociada al cliente, que permite la autenticación segura en el consumo del servicio.

Estas credenciales deberán ser almacenadas y gestionadas de forma segura por el equipo de desarrollo o infraestructura de la entidad, y serán utilizadas en los puntos de autenticación definidos por ID Perú, por tanto, todo el tráfico de solicitudes generado mediante estas credenciales estará directamente vinculado a la entidad integradora, siendo por tanto de su exclusiva responsabilidad.

5. MECANISMOS DE SEGURIDAD

El servicio de ID Perú implementa múltiples controles de seguridad para proteger el acceso a sus recursos, garantizar la confidencialidad de las transacciones y asegurar la integridad de cada proceso de autenticación. Entre los mecanismos más relevantes se incluyen:

- ✓ Control de acceso basado en convenio: Solo las entidades que cuenten con un convenio vigente con RENIEC y hayan recibido credenciales oficiales (“***Client_ID***” y “***Client_Secret***”) pueden acceder a los servicios del sistema y solo para los fines que en el convenio vigente estén establecidos.
- ✓ Transmisión segura mediante HTTPS: Todo el intercambio de datos (solicitudes, respuestas y tokens) se realiza a través del protocolo HTTPS, asegurando el cifrado de extremo a extremo y protegiendo contra ataques de tipo man-in-the-middle (MITM).

- ✓ Protección de integridad mediante sello de tiempo (TimeStamp): Las respuestas generadas por el servicio (especialmente los tokens de autenticación) incluyen un sello de tiempo digital RFC 3161, que permite a la entidad consumidora verificar la integridad y validez temporal de la transacción autenticada.

  Estos mecanismos, combinados, aseguran que solo actores autorizados interactúen con el sistema y que cada autenticación sea verificable, íntegra y protegida contra modificaciones no autorizadas.

6. FLUJO DE AUTENTICACIÓN ID PERÚ

Este apartado describe el flujo de autenticación implementado en la plataforma ID Perú, como el uso de códigos QR y verificación biométrica. Esta combinación permite autenticar de forma inequívoca la identidad de una persona natural, elevando los niveles de seguridad, usabilidad y trazabilidad en los procesos digitales.

1. Requisitos Operativos.

Para llevar a cabo este flujo de autenticación reforzada, se requiere:

a. Un dispositivo móvil (smartphone) con cámara funcional y sistema operativo Android o iOS.
b. Conectividad a Internet estable (Wi-Fi o datos móviles).
c. Lector biométrico en el dispositivo dactilar o lector de DNIe, según el método de autenticación configurado.
d. Contar con las credenciales de acceso ("***Client_ID***" y "***Client_Secret***") previamente habilitadas y registradas en el sistema.

Cabe señalar que los elementos descritos anteriormente —como el dispositivo móvil, el lector biométrico y la conectividad a Internet— constituyen herramientas indispensables para habilitar el proceso de identificación reforzada. Sin embargo, no están limitadas a un único método de autenticación, ya que la plataforma ID Perú permite su reutilización en distintos flujos y niveles de autenticación.

2. Archivo de Configuración de Autenticación.

Una vez cumplidos los requisitos operativos, es necesario contar con una lista de parámetros esenciales para inicializar correctamente el proceso de autenticación contra la plataforma ID Perú. Estos parámetros pueden estar ubicados o implementados en el formato que mejor se adapte al entorno del sistema (por ejemplo: .json, .properties, .yaml o .env), según el lenguaje de programación o framework utilizado. La estructura y ubicación del archivo quedan a criterio del equipo de desarrollo, siempre que garantice su correcto uso al momento de iniciar el flujo de autenticación.

A continuación, se detallan los parámetros obligatorios que se necesitaran para realizar un proceso de solicitud de identificación:

PARÁMETRO	DESCRIPCIÓN TÉCNICA	TIPO / VALOR ESPERADO
client_id	Identificador único del cliente otorgado por RENIEC para cada proyecto. Se entrega por canal oficial.	string
client_secret	Clave secreta asociada al client_id, generada por RENIEC y enviada junto con las credenciales.	string
auth_uri	URL fija del servicio de autenticación de RENIEC. Utilizada para iniciar el flujo de autenticación (/auth).	https://idaas.reniec.gob.pe /service/auth
token_uri	URL fija del servicio para intercambio de código de autorización por token (/token).	https://idaas.reniec.gob.pe /service/token
userinfo_uri	URL del servicio que retorna los datos del usuario autenticado bajo demanda (/userinfo).	https://idaas.reniec.gob.pe /service/userinfo
logout_uri	URL del servicio de cierre de sesión (/logout).	https://idaas.reniec.gob.pe /service/logout
auth_provider _keys_uri	URL para obtener las llaves públicas del proveedor de autenticación (JWKs).	https://idaas.reniec.gob.pe /service/certs
redirect_uri	URL segura (HTTPS) a la que será redirigido el usuario luego de completar el proceso de identificación.	string (URL registrada)

Tabla 1: Parámetros para un proceso de solicitud de Identificación.

Ejemplo de los parámetros en un archivo de configuración en formato JSON.

```
{
    "client_id" : "FhPhzs7CNWvq0b9rFo9Rh9kQYnN",
    "client_secret" : "ekNEb8AbNp6JcXusZ3mFA",
    "auth_uri": "https://idaas.reniec.gob.pe/service/auth",
    "token_uri": "https://idaas.reniec.gob.pe/service/token",
    "userinfo_uri": "https://idaas.reniec.gob.pe/service/userinfo",
    "logout_uri": "https://idaas.reniec.gob.pe/service/logout",
    "auth_provider_keys_uri": "https://idaas.reniec.gob.pe/service/certs",
    "redirect_uris": ["https://miapp.entidad.gob.pe/auth/callback",
    "https://miapp.entidad.gob.pe/home"]
}
```

Código Fuente 1: Ejemplo de archivo de configuración.

Ventajas de un archivo de configuración

- ✓ Facilita el mantenimiento del sistema al permitir la actualización de parámetros sin alterar el código fuente.
- ✓ Proporciona una estructura clara y reutilizable entre entornos (desarrollo, QA, producción).
- ✓ Permite gestionar cambios operativos, como la rotación de credenciales (“***client_id***”, “***client_secret***”) o actualizaciones en los “endpoints, de forma sencilla.

Alternativas válidas. La institución integradora podrá utilizar cualquier otro mecanismo que considere conveniente para manejar estos valores, siempre que:

- ✓ Se garantice la seguridad y confidencialidad de las credenciales.
- ✓ Los valores utilizados cumplan con los formatos y requisitos definidos por RENIEC.
- ✓ La integración final mantenga compatibilidad plena con los flujos y parámetros exigidos por ID Perú.

Esto incluye, por ejemplo, el uso de:

- ✓ Variables de entorno (environment variables).
- ✓ Configuración desde base de datos.
- ✓ Almacenamiento en servicios secretos o cofres de claves (Key Vaults).
- ✓ Carga dinámica desde servicios de configuración remotos (config servers).

3. Proceso de creación de la URL de solicitud de Identificación Ciudadano

Para esta forma en concreto se contará con un archivo de configuración (Código Fuente 1: Ejemplo de archivo de configuración.) en el cual se almacenarán lo parámetros previamente definidos; el proceso de solicitud de identificación se concreta con la solicitud al endpoint "***/auth***", con los parámetros debidamente establecidos para ello se debe construir una URL de solicitud que permita iniciar el flujo de autenticación del ciudadano, conforme a las especificaciones de OpenID Connect.

Esta URL utiliza como base el siguiente Authorization Endpoint oficial de ID Perú: Ruta de Servicio.

```
https://idaas.reniec.gob.pe/service/auth
```

A este endpoint se deben anexar los siguientes parámetros los cuales se enviará por método GET, codificados adecuadamente como parte de la cadena de solicitud:

PARÁMETRO	DESCRIPCIÓN TÉCNICA
response_type	Define el tipo de respuesta esperada por el cliente
client_id	Identificador único proporcionado por RENIEC a la entidad integradora.
redirect_uri	URL segura (HTTPS) a la que se redirigirá al usuario tras autenticarse correctamente. Debe coincidir con la registrada.
state	Código aleatorio generado por el cliente para evitar ataques CSRF.
scope	Define el nivel de acceso solicitado. Los valores permitidos son: openid y profile (separados por espacio).
vd	Valor del DNI del ciudadano (aplicable solo a mayores de edad).
acr_values	Determina el flujo de autenticación que será utilizado: - two_factor: Autenticación con dos factores: uso de One-Time Password (OTP) por SMS o correo electrónico, junto con la autenticación biométrica facial mediante la aplicación móvil ID Perú. El registro del número móvil o correo electrónico se

	realiza previamente en la plataforma ID Perú a través de la biometría facial. - face_mobile: Autenticación biométrica facial mediante el uso de la aplicación móvil ID Perú. - pki_dnie: Autenticación utilizando el certificado digital de autenticación del DNI electrónico. Requiere Windows 8 o una versión superior, así como NET Framework 4.5 o superior. El proceso de autenticación se lleva a cabo en la PC o laptop. - pki_token: Autenticación utilizando el certificado de persona jurídica instalado en el sistema operativo o dispositivo criptográfico. Requiere Windows 8 o una versión superior, así como NET Framework 4.5 o superior. El proceso de autenticación se lleva a cabo en PC o laptop. - pki_dnie_legacy: Autenticación utilizando el certificado digital de autenticación del DNI electrónico. Requiere Windows 8 o una versión superior, así como Java JRE 8 actualizado. El proceso de autenticación se lleva a cabo en PC o laptop. - pki_token_legacy: Autenticación utilizando el certificado de persona jurídica instalado en el sistema operativo o dispositivo criptográfico. Requiere Windows 8 o una versión superior, así como Java JRE 8 o superior. El proceso de autenticación se lleva a cabo en PC o laptop. - fingerprint_mobile: Autenticación biométrica dactilar mediante el uso de la aplicación móvil ID Perú. - one_factor: Autenticación con un solo factor, mediante el uso de una contraseña creada previamente mediante autenticación biométrica facial en la plataforma ID Perú.

Tabla 2: Descripción de parámetros para la solicitud de identificación

Ejemplo ilustrativo de los parámetros anteriormente mencionados.

```
https://idaas.reniec.gob.pe/service/auth?
response_type=code&
client_id=abc123xyz&
redirect_uri=https%3A%2F%2Fmiapp.entidad.gob.pe%2Fauth%2Fcallback&
state=ZGF0b1ZlcnNpZmljYWRv&
scope=openid+profile+offline_access&
vd=12345678&
acr_values=two_factor
```

Código Fuente 2: URL Ilustrativo de Solicitud de Identificación

El manejo de cada uno de los parámetros utilizados en la URL de solicitud de autenticación puede variar según el lenguaje de programación, entorno de ejecución, o arquitectura de la aplicación integradora (back-end, front-end, cliente web, móvil, etc.). No obstante, el resultado final debe cumplir con la estructura y formato requerido por la plataforma ID Perú, respetando las validaciones establecidas por el endpoint de autorización.

A continuación, se detalla el tratamiento individual de cada parámetro y su construcción paso a paso. Este proceso puede ser implementado mediante funciones específicas, objetos de configuración o construcción manual de la URL, siempre que el resultado final sea correcto y cumpla los estándares de

codificación de URI. Al finalizar esta sección, se mostrará un ejemplo completo de URL correctamente construida como referencia práctica para validación o pruebas de integración.

A. Parámetro (acr_values).

El parámetro acr_values define de forma explícita y obligatoria el tipo de autenticación que debe ejecutarse durante el flujo de identificación del usuario. Este valor condiciona directamente la lógica de presentación, los factores de autenticación disponibles son.

- ✓ two_factor
- ✓ face_mobile
- ✓ pki_dnie
- ✓ pki_token
- ✓ pki_dnie_legacy
- ✓ pki_token_legacy
- ✓ fingerprint_mobile
- ✓ one_factor

Consideraciones Técnicas.

- ✓ Obligatorio: Este parámetro debe ser incluido en toda URL de solicitud de identificación.
- ✓ Sensitivo a mayúsculas/minúsculas: Los valores deben ser escritos exactamente como se especifican, en minúsculas y sin espacios.
- ✓ Exclusividad: Solo puede establecerse un valor por solicitud. No se permite enviar múltiples métodos en una misma URL.

Ejemplo de uso (URL):

```
...&acr_values=face_mobile
```

Código Fuente 3: parámetro acr_values

B. Parámetro (client_id).

El parámetro client_id representa la credencial única de identificación de la entidad integradora, asignada por RENIEC durante el proceso de registro y validación institucional. Esta credencial es obligatoria para toda solicitud de autenticación y debe ser utilizada sin modificación.

Especificaciones Técnicas

- ✓ Origen: Este valor es generado y proporcionado exclusivamente por RENIEC a través del canal oficial.
- ✓ Formato: Cadena alfanumérica de longitud fija de 27 caracteres.
- ✓ Codificación: Internamente codificado por RENIEC. No es necesario, ni permitido, alterarlo o transformarlo.

Debe ser insertado en la URL tal como fue entregado, es decir, en texto plano y sin codificación adicional (como Base64 o URI encode), excepto si lo exige el contexto de construcción de URL.

Consideraciones Técnicas.

- ✓ No modificar ni recortar el valor original.
- ✓ Este valor se encuentra en el archivo de configuración mostrado anteriormente.
- ✓ No aplicar codificación adicional, salvo la necesaria para construir la URL completa correctamente (application/x-www-form-urlencoded).
- ✓ No reutilizable entre proyectos: Cada client_id es específico para un entorno o solución registrada.

Ejemplo de uso (URL):

```
...&client_id=ABC123456789XYZ987654321qwe
```

Código Fuente 4: parámetro client_id

C. Parámetro (response_type).

El parámetro "***response_type***" indica el tipo de respuesta que el cliente espera recibir al finalizar correctamente el flujo de autenticación en la plataforma ID Perú. Su valor determina el comportamiento del servidor de autorización frente a la solicitud iniciada.

Especificaciones Técnicas

- ✓ Valor obligatorio: code
- ✓ Función: Solicita un "***authorization code***" temporal como respuesta inicial del flujo de identificación, el cual será posteriormente intercambiado por un "***access token***" y un "***id token***".

Consideraciones Técnicas.

- ✓ Este valor debe establecerse exactamente como:

Ejemplo de uso (URL):

```
...&response_type=code
```

Código Fuente 5:parámetro response_type

D. Parámetro (redirect_uri).

El parámetro "***redirect_uri***" define la URL del cliente a la que el servicio ID Perú redirigirá al usuario una vez completado el proceso de autenticación, ya sea exitoso o fallido. Es el punto de retorno donde el sistema integrador recibirá los resultados de la identificación (como el code, state, o un posible error).

Especificaciones Técnicas.

- ✓ Tipo de dato: URL absoluta (esquema https:// obligatorio).

- ✓ Función: Actuar como callback endpoint para continuar el flujo de identificación del ciudadano desde el lado del cliente.
- ✓ Seguridad: La URL debe estar previamente registrada ante RENIEC como parte del proceso de integración. Solo se aceptarán redirecciones hacia URLs autorizadas.
- ✓ Codificación Requerida: Debido a que el parámetro "***redirect_uri***" es una URL pasada como valor dentro de otra URL (la de autenticación), debe estar codificada usando percent-encoding (también conocida como URL encoding), conforme a las reglas de codificación de application/x-www-form-urlencoded.

Ejemplo.

```
: se convierte en %3A
/ se convierte en %2F
```

Esto evita errores en la construcción de la URL final y asegura que el endpoint de redirección sea interpretado correctamente por el servidor.

Ejemplo de codificación.

```
//URL para endpoint de la aplicación que se está integrando.
https://miapp.entidad.gob.pe/auth/callback
//URL codificada en URL Encoding
https%3A%2F%2Fmiapp.entidad.gob.pe%2Fauth%2Fcallback
```

Código Fuente 6: URL encoding

Consideraciones técnicas.

- ✓ Aunque el parámetro "***redirect_uris***" dentro del archivo de configuración puede estar definido como un arreglo de múltiples URLs (por ejemplo, para entornos como desarrollo, pruebas y producción), al momento de construir la solicitud de autenticación, solo se debe utilizar una única URL. La plataforma ID Perú redirigirá la respuesta exclusivamente al valor especificado en el parámetro "***redirect_uri***" incluido en la URL de solicitud. Este debe coincidir exactamente con uno de los valores registrados previamente en RENIEC.

Ejemplo de uso (URL):

```
...&redirect_uri=https%3A%2F%2Fmiapp.entidad.gob.pe%2Fauth%2Fcallback
```

Código Fuente 7: parámetro redirect_uri

E. Parámetro (state).

El parámetro state es una cadena de caracteres generada por la plataforma integradora (cliente) y enviada como parte de la URL de solicitud de autenticación. Su función principal es actuar como token de correlación para validar la integridad de la sesión de autenticación y proteger contra ataques de tipo Cross-Site Request Forgery (CSRF).

Especificaciones técnicas.

- ✓ Tipo de dato: string (cadena alfanumérica codificada).

- ✓ Generación: Debe ser único por cada sesión de autenticación y de un solo uso.
- ✓ Recomendación: Para garantizar su unicidad sin necesidad de algoritmos complejos, se recomienda generar el valor utilizando el timestamp actual en formato Unix (epoch time).
- ✓ Codificación:
  - El valor debe ser codificado en Base64, ya que el servicio ID Perú espera decodificarlo internamente.
  - Adicionalmente, debe aplicarse percent-encoding (URL encoding), especialmente porque la codificación Base64 puede incluir caracteres reservados como =, +, /.

Consideraciones técnicas.

- ✓ El "***state***" debe ser generado por la institución integradora.
- ✓ Aunque se recomienda usar el tiempo Unix como valor base, la entidad puede implementar otro esquema de generación que garantice unicidad (UUID, hash, etc.).
- ✓ El valor recibido por ID Perú será decodificado automáticamente, por lo tanto, es fundamental que la codificación inicial sea precisa.

Ejemplo en pseudocódigo

```
timestamp = getUnixTimestamp()
// Ej: 1716149873
state_raw = toString(timestamp)
state_base64 = base64Encode(state_raw)
// Ej: MTcxNjE0OTg3Mw==
state_encoded = urlEncode(state_base64)
// Ej: MTcxNjE0OTg3Mw%3D%3D
```

Código Fuente 8: Ejemplo de generación de parámetro state

Ejemplo de uso (URL):

```
...&state=MTcxNjE0OTg3Mw%3D%3D
```

Código Fuente 9: Parámetro state

F. Parámetro (vd).

El parámetro vd corresponde al valor del Documento Nacional de Identidad (DNI) de la persona natural a la que se desea autenticar mediante la plataforma ID Perú. Este dato es sensible y está directamente asociado a la identidad del ciudadano, por lo que su transmisión debe realizarse bajo condiciones especiales de cifrado seguro. Su propósito principal es permitir que la plataforma ID Perú conozca de antemano el número de DNI del ciudadano objetivo de la autenticación, de modo que el proceso pueda iniciarse directamente sobre esa identidad.

Dado que el "***vd***" contiene información de carácter personal, no debe ser enviado en texto plano. En su lugar, debe ser cifrado utilizando el algoritmo AES bajo las siguientes condiciones técnicas:

PROPIEDAD	VALOR REQUERIDO
Algoritmo de cifrado	AES (Advanced Encryption Standard)
Modo de operación	CBC (Cipher Block Chaining)
Relleno (padding)	PKCS5
Codificación final	Base64 (del resultado cifrado)
Transmisión	URL encoded del valor Base64

Tabla 3: Propiedades del parámetro vd

Especificaciones técnicas

- ✓ Cifrado del DNI (vd) para su Envío Seguro. Para enviar el número de DNI del ciudadano de manera segura como parte de la URL de solicitud de autenticación, la plataforma requiere que este valor sea encriptado utilizando el algoritmo AES en modo CBC con relleno PKCS5. Esta técnica de cifrado simétrico garantiza la confidencialidad del dato sensible durante la transmisión.

  Requisitos Técnicos del Cifrado:
  - Algoritmo: AES (Advanced Encryption Standard)
  - Modo de operación: CBC (Cipher Block Chaining)
  - Relleno: PKCS5
  - Codificación final: Base64 + URL encoding
- ✓ Parámetros de entrada para la función de cifrado. Se debe construir un método específico que reciba los siguientes parámetros:
  - dni: Número de DNI del ciudadano (cadena de 8 dígitos).
  - client_id: Valor del Client ID proporcionado por RENIEC (cadena de 27 caracteres) y que se puede extraer del archivo de configuración.
- ✓ Derivación de elementos criptográficos ambos valores requeridos por el algoritmo AES se derivan del client_id:
  - IV (Initialization Vector): Primeros 16 caracteres del client_id.
  - KEY (clave de cifrado): También se obtienen los primeros 16 caracteres del client_id.

Ejemplo en seudocódigo.

```
FUNC encryptDNI(dni: STRING, client_id: STRING) RETURNS STRING:
    iv = client_id.substring(0, 16)
    key = client_id.substring(0, 16)

    // Convertir DNI a bytes
    inputBytes = stringToBytes(dni)

    // Inicializar AES en modo CBC con padding PKCS5
    cipher = AES.new(key, mode = CBC, iv = iv, padding = PKCS5)

    // Encriptar los datos
    encryptedBytes = cipher.encrypt(inputBytes)
```

```
// Codificar en Base64
base64Encrypted = base64Encode(encryptedBytes)

// Aplicar codificación URL (percent-encoding)
urlEncoded = urlEncode(base64Encrypted)

RETURN urlEncoded
```

Código Fuente 10 : Método para encriptar parámetro vd

- ✓ El valor devuelto por esta función debe ser usado como el valor del parámetro "***vd***" en la construcción de la URL de autenticación.

Ejemplo de uso (URL):

```
…&vd=sOJR4PLPbFCCqlTMAmOZrw%3D%3D
```

Código Fuente 11: Parámetro vd

Consideraciones técnicas

- ✓ El cifrado del número de DNI como parte del proceso de autenticación no solo es obligatorio, sino que debe seguir con precisión las reglas de derivación y encriptación definidas.
- ✓ Derivación determinista. La derivación del "***IV***" y de la "***KEY***" debe ser completamente determinista y reproducible, es decir, el resultado de la función debe ser exactamente el mismo para una combinación dada de DNI y Client ID. Esto es esencial para que los servidores de ID Perú puedan desencriptar correctamente el valor recibido en el parámetro "***vd***".
- ✓ Uniformidad en todos los entornos. El método de encriptación debe estar implementado de forma idéntica en todos los entornos: desarrollo, pruebas (QA) y producción. Se recomienda realizar pruebas cruzadas entre cliente y servidor para validar que la encriptación/desencriptación sea compatible y funcional.
- ✓ Posibles cambios en el esquema criptográfico. Aunque actualmente la derivación de "***IV***" y "***KEY***" se basa en los primeros 16 caracteres del "***client_id***", este esquema puede ser modificado en el futuro por parte de RENIEC. Cualquier modificación será notificada a la entidad integradora a través de canales oficiales y seguros, y deberá ser implementada respetando los estándares que la propia RENIEC indique por medios oficiales.
- ✓ Impacto en el proceso de autenticación. El cifrado correcto del parámetro "***vd***" es crítico para la validación del flujo de autenticación. Si el valor recibido por el servidor no puede ser desencriptado correctamente, o si el contenido desencriptado no coincide con un DNI válido, el servicio ID Perú rechazará la solicitud y responderá con un error general de autenticación. Este tipo de error no indica una falla en el servicio, sino un error en la construcción o codificación de la solicitud por parte del cliente.

G. Parámetro (scope).

El parámetro scope define el nivel de información del usuario final que la plataforma integradora desea obtener como parte del proceso de autenticación. Esta solicitud debe alinearse estrictamente con lo establecido en el convenio vigente entre la entidad integradora y RENIEC.

Especificaciones técnicas.

- ✓ Tipo de dato: string (cadena de valores separados por espacio).
- ✓ Función: Solicitar autorización explícita para acceder a la informacion del ciudadano identificado.
- ✓ Valores válidos actualmente:

VALOR	DESCRIPCIÓN
openid	Requerido. Activa el flujo de autenticación OpenID.
profile	Solicita acceso a información básica del ciudadano (nombre, sub, etc.).

Tabla 4: propiedades del parámetro scope

- ✓ El valor "*openid*" **es obligatorio** y debe estar presente en toda solicitud, ya que habilita la identificación bajo OpenID Connect.
- ✓ Formato requerido. Los valores deben ir separados por un único espacio en blanco. La cadena debe ser codificada usando URL encoding (application/x-www-form-urlencoded), ya que se incluirá en una URL. El parámetro es sensible a mayúsculas/minúsculas, por lo que se deben escribir los valores exactamente como se indica (openid, no OpenID, por ejemplo).

Consideraciones técnicas.

- ✓ Aunque es técnicamente posible incluir valores en el parámetro "***scopes"***, que no estén autorizados por convenio o que sean inválidos, el servicio de ID Perú simplemente los ignorará. No generarán error, pero tampoco serán procesados ni devueltos.
- ✓ Se recomienda incluir únicamente los valores del parámetro "***scopes***" habilitados para evitar procesos internos innecesarios.
- ✓ Errores comunes como uso incorrecto de mayúsculas, dobles espacios o uso de caracteres inválidos pueden resultar en fallos de validación o en el rechazo silencioso del parámetro.

Ejemplo de uso (URL):

```
…&scope=openid+profile
```

Código Fuente 12: Parámetro scope

4. Integración Final: Construcción de la URL de Solicitud de Autenticación

Una vez que todos los parámetros han sido procesados y codificados conforme a las especificaciones técnicas, se procede a ensamblar la URL final para realizar

la solicitud de autenticación contra la plataforma de ID Perú. Esta URL será utilizada por el navegador o aplicación cliente para redirigir al usuario al flujo de autenticación.

A. Archivo de configuración.

Aunque en el presente documento se propone el uso de un archivo de configuración estructurado (por ejemplo, en formato JSON) para centralizar los valores necesarios en el proceso de autenticación —como el client_id, client_secret, URIs del servicio y el redirect_uri—, su implementación no es estrictamente obligatoria.

Archivo de configuración (reniec_idaas.json).

```
{
 "client_id": "abc123456789xyz987654321qwe",
 "client_secret": "secr3tP@ssw0rd",
 "auth_uri": "https://idaas.reniec.gob.pe/service/auth",
 "token_uri": "https://idaas.reniec.gob.pe/service/token",
 "userinfo_uri": "https://idaas.reniec.gob.pe/service/userinfo",
 "logout_uri": "https://idaas.reniec.gob.pe/service/logout",
 "auth_provider_keys_uri":
"https://idaas.reniec.gob.pe/service/certs",
 "redirect_uris": [
    "https://miapp.entidad.gob.pe/auth/callback"
  ]
}
```

Código Fuente 13: Muestra de archivo de configuración

B. URL Construida (ejemplo con todos los parámetros)

```
https://idaas.reniec.gob.pe/service/auth?
response_type=code&
client_id=abc123456789xyz987654321qwe&
redirect_uri=https%3A%2F%2Fmiapp.entidad.gob.pe%2Fauth%2Fcallback&
state=MTcxNjIwNTQ1NQ%3D%3D&
scope=openid+profile&
vd=6nA1l5pP5h7F0D0zR3xw%3D%3D&
acr_values=fase_mobile
```

Código Fuente 14: URL de solicitud de identificación final

C. Resumen tecnico de cada parámetro integrado.

A continuación, un resumen practico de cada parámetro:

PARÁMETRO	ORIGEN	DESCRIPCIÓN
response_type	Fijo	Tipo de respuesta: code
client_id	Archivo de configuración	Identificador único del cliente
redirect_uri	Archivo de configuración	URL de retorno después de autenticación en URL Encoding.
state	Generado por cliente	Token único de correlación de sesión en Base64
scope	Definido por el cliente	Niveles de acceso solicitados (openid, profile, etc.)
vd	DNI cifrado AES + Base64 + URL	Identificador cifrado del ciudadano a autenticar

acr_values	Definido por cliente	Tipo de autenticación requerida definir uno de la lista. - two_factor - face_mobile: - pki_dnie - pki_token - pki_dnie_legacy - pki_token_legacy - fingerprint_mobile - one_factor ()

Tabla 5: Resumen de parámetros de la URL de solicitud de identificación

7. ENVIO DE URL DE SOLICITUD.

Una vez construida la URL de autenticación con todos los parámetros requeridos, el siguiente paso es enviar la solicitud HTTP por medio del método GET al servidor de autenticación de RENIEC.

Este paso puede ser ejecutado mediante una redirección desde el navegador del usuario o desde una aplicación cliente que maneje sesiones autenticadas.

Método de envío

- ✓ Tipo de solicitud: GET
- ✓ Destino: https://idaas.reniec.gob.pe/service/auth
- ✓ Canal: HTTP seguro (HTTPS)
- ✓ Responsable: Cliente integrador (sistema web, app móvil, navegador, etc.)

Proceso de respuesta

Una vez que el servidor de ID Perú reciba y valide la solicitud, redirigirá al usuario al flujo de autenticación correspondiente (biometría, OTP, DNIe, etc.). Al finalizar dicho proceso, el servidor redirigirá nuevamente a la dirección especificada en "***redirect_uri***" del cliente, incluyendo los siguientes parámetros:

- ✓ code: Código de autorización temporal (si la autenticación fue exitosa).
- ✓ state: Valor enviado originalmente por el cliente (para validación).
- ✓ error: Solo presente si ocurrió una falla (cancelación, timeout, error interno, etc.).

Gestión de Sesión y Correlación con el Parámetro "state"

Importancia del Manejador de Sesión: Este proceso debe realizarse dentro del contexto de una sesión HTTP activa, gestionada por la plataforma cliente (integradora). La sesión es fundamental para mantener el estado del usuario entre la solicitud inicial y la respuesta de redirección que será enviada por RENIEC una vez concluido el flujo de autenticación.

Correlación mediante "***state***": Para garantizar la integridad y trazabilidad del proceso, se debe establecer una asociación explícita entre la sesión activa y el parámetro "***state***" enviado en la URL de solicitud. Este valor, generado previamente por el cliente, actúa como token único de identificación de la solicitud. Se recomienda guardar el valor del parámetro "***state***" en la sesión activa ("***session.state = generado***") antes de redirigir al usuario a la URL de autenticación. De esta forma, cuando el usuario sea redirigido de regreso a la dirección URL del tipo endpoint establecida en el parámetro "***redirect_uri***" con los parámetros de respuesta (code, state, etc.), el sistema podrá:

- ✓ Verificar que el valor del parámetro "***state***" recibido coincide con el que fue almacenado en sesión.
- ✓ Confirmar que la respuesta corresponde efectivamente a la solicitud inicial.
- ✓ Evitar ataques tipo CSRF (Cross-Site Request Forgery) u otros intentos de suplantación.

Ejemplo en seudocódigo (Backend): generación de "***state***" y envio de URL.

```
// Paso 1: Generar el parámetro 'state'
state = base64Encode(getUnixTimestamp())
session["auth_state"] = state

// Paso 2: Redirigir al usuario a la URL de autenticación
authUrl = construirUrlConParametros(state, otrosParametros)
redirectTo(authUrl)
```

Código Fuente 15: Generación parámetro state

Ejemplo en seudocódigo (Backend): comparación de "***state***" con "***sesiónstate***".

```
// Paso 3: Al recibir la redirección de vuelta en redirect_uri
IF session["auth_state"] == request.getParam("state") THEN
    continuarProceso()
ELSE
    rechazarPeticion("State inválido o expirado")
```

Código Fuente 16: verificación parámetro state

- ✓ Este mecanismo de correlación garantiza una autenticación segura y confiable para ambas partes: la plataforma cliente y el servicio ID Perú.

8. PROCESO DE IDENTIFICACIÓN

Una vez que la URL de solicitud de identificación ha sido correctamente construida y enviada al servicio ID Perú de RENIEC, el sistema del usuario final es redirigido automáticamente al flujo de autenticación correspondiente, según el valor definido en el parámetro "***acr_values***" (Tabla 2: Descripción de parámetros para la solicitud de identificación).

¿Qué sucede en esta etapa?

El servicio ID Perú inicia el proceso de identificación y autenticación del ciudadano, el cual puede variar dependiendo del método de identificación elegido. Este proceso

es completamente gestionado por los servicios de RENIEC y no requiere intervención directa del cliente integrador, salvo haber solicitado el flujo correcto y valido que se encuentre establecido en convenio vigente con la institución.

Consideraciones Técnicas

- ✓ Este proceso puede involucrar distintos dispositivos del usuario: celular, lector de tarjetas inteligentes, lector de huellas digitales, navegador web o la app ID Perú.
- ✓ La experiencia del usuario y el tiempo de respuesta dependen del flujo seleccionado y del canal de validación biométrica habilitado sin descartar la velocidad en la conexión a internet.
- ✓ La aplicación integradora no interviene directamente en esta etapa; simplemente espera la redirección final con el resultado.

Resultado de esta etapa

Al finalizar el proceso (ya sea exitoso o fallido), el servicio ID Perú redirige al usuario a la dirección URL del tipo endpoint que se estableció en el parámetro "***redirect_uri***" registrado por el cliente, el mismo se encuentra en el archivo de configuración.

9. PROCESO DE RECEPCIÓN Y LECTURA DE LA RESPUESTA DE ID PERÚ

Una vez completado el flujo de autenticación por parte del ciudadano en la plataforma ID Perú, el sistema redirige automáticamente al usuario hacia el endpoint definido por la entidad integradora bajo el parámetro "***redirect_uri***". Este endpoint, registrado previamente ante RENIEC y definido en nuestro caso en el archivo de configuración (Código Fuente 13: Muestra de archivo de configuración

), será invocado por medio de una solicitud HTTP de tipo GET, conteniendo los parámetros necesarios para continuar el proceso de identificación.

```
https://miapp.entidad.gob.pe/auth/callback
```

Código Fuente 17: Dirección URL endpoint del parámetro redirect_uri

Este valor debe coincidir exactamente con el registrado en RENIEC y codificado correctamente en la URL de autenticación.

Parámetros incluidos en la redirección. La solicitud enviada por ID Perú hacia la dirección URL del tipo endpoint establecida en el parámetro "***redirect_uri***" incluirá los siguientes parámetros sobre la URL:

PARÁMETRO	DESCRIPCIÓN
code	Código de autorización temporal. Será utilizado posteriormente para obtener los tokens de acceso e identificación.
state	Valor único enviado originalmente por el cliente. Debe ser validado para garantizar la integridad de la sesión.

error	Solo presente si ocurrió un fallo en la autenticación. Puede indicar cancelación, expiración o error técnico.

Tabla 6: Parámetros de la URL de respuesta

Además de procesar los parámetros "***code***", "***state***" y "***error***", es fundamental que la aplicación integradora valide el valor del parámetro "***session state***", es decir, el valor que fue almacenado previamente en la sesión HTTP del usuario cuando se construyó y se envió la solicitud de identificación ciudadana. Si el valor del parámetro "***state***" recibido no coincide con el valor del parámetro "***session state***", la solicitud debe ser considerada inválida y rechazada inmediatamente por la aplicación integradora.

Observaciones técnicas.

- ✓ La validación del parámetro "***state***" con respecto al "***sesión_state***" es crítica para prevenir ataques de tipo CSRF.
- ✓ Toda esta lógica debe implementarse por la aplicación integradora correspondiente al backend correspondiente a la URL establecida en el parámetro "***redirect_uri***".
- ✓ En caso de error o incongruencia en los datos recibidos, el proceso debe redirigir al usuario a una ruta segura o mostrar un mensaje controlado.

Pseudocódigo del proceso de recepción y validación

```
FUNC manejarRespuestaDeIdPeru(request):
//Extraer parámetros de la URL
 error = request.getParam("error")
 code = request.getParam("code")
 state = request.getParam("state")

 // Obtener el valor almacenado en sesión
 sessionState = request.getSessionAttribute("state")

 // Validar existencia y coherencia de los datos
 IF error IS NULL AND code IS NOT NULL:
  IF state == sessionState:

   cliente = crearclienteReniec()
   tokenResponse = cliente.obtenerTokens(code)

   IF tokenResponse IS NOT NULL:
    userInfo = cliente.obtenerUserInfo(tokenResponse.access_token)
    session.setAttribute("usuario", userInfo)
    LOG("Usuario autenticado: " + userInfo)
       logicaDeNegocio(userInfo)
    REDIRIGIR_A vistaPrincipal()
   ELSE:
    LOG_ERROR("Validación fallida: state no coincide con sessionState")
 ELSE:
  LOG_WARN("Error recibido de ID Perú: " + error)

 REDIRIGIR_A vistaControlada()
```

Código Fuente 18: Manejo de respuesta desde la URL del parámetro redirect_uri

1. VALIDANDO ERRORES Y PARÁMETROS CRITICOS.

Al recibir la redirección desde el servicio ID Perú hacia la URL establecida en el parámetro "***redirect_uri***" perteneciente a la aplicación integradora, lo primero que

debe realizarse es la validación de los parámetros retornados, asegurando la legitimidad de la respuesta antes de continuar con el flujo.

Parámetros a validar

“error”

- ✓ Tipo: String (opcional)
- ✓ Validación esperada: Debe ser null o no estar presente.
- ✓ Significado: Si este parámetro está presente, indica que el proceso de autenticación no se completó correctamente.
- ✓ Acción recomendada: Si existe, se debe detener el flujo de autenticación inmediatamente, registrar el error y redirigir al usuario a una vista controlada o de error.

“code”

- ✓ Tipo: String (obligatorio si el parametro “***error***” tiene el valor esperado).
- ✓ Validación esperada: Debe contener una cadena alfanumérica válida que puede incluir los caracteres - y _.
- ✓ Uso: Este valor es esencial para realizar el intercambio por tokens (/token). Su ausencia invalida el flujo.
- ✓ Acción recomendada: Si el parámetro está vacío o es inválido, detener el flujo y registrar el incidente.

“state”

- ✓ Tipo: String (obligatorio si el parámetro “***code***” es válido)
- ✓ Validación esperada: Debe coincidir exactamente con el valor almacenado previamente en la sesión HTTP activa (session["state"]).
- ✓ Propósito: Asegura la integridad de la sesión y previene ataques CSRF (Cross-Site Request Forgery).
- ✓ Acción recomendada: Si no coincide, el proceso debe ser abortado inmediatamente y marcarse como intento no válido.

2. INTERCAMBIO DEL CODIGO DE AUTORIZACIÓN POR TOKEN.

Una vez validado correctamente el parámetro “***code***” y asegurada la integridad del parámetro “***state***” lo cual establece que es la respuesta a la solicitud enviada originalmente, el siguiente paso en el flujo de autenticación es realizar la solicitud del token de acceso al servidor de autorización de ID Perú. Este proceso se lleva a cabo mediante una solicitud HTTP de tipo POST al endpoint que se encuentra en el archivo de configuración con el parámetro “***token_uri***” (Código Fuente 13: Muestra de archivo de configuración

).

Importancia del parámetro "***code***".

- ✓ El parámetro "***code***" recibido es un código de autorización temporal, generado por RENIEC únicamente para esa sesión.
- ✓ Este código es de un solo uso y tiene una vida útil limitada (típicamente en segundos o minutos).
- ✓ Si el código es mal utilizado, expirado o se intenta reutilizar, el servidor rechazará la solicitud con un error.
- ✓ Por tanto, su tratamiento correcto es fundamental para completar con éxito el proceso de identificación.

Especificaciones para la solicitud al enpoint ***"/token***".

- ✓ Método HTTP: POST
- ✓ URL: https://idaas.reniec.gob.pe/service/token
- ✓ Content-Type: application/x-www-form-urlencoded
- ✓ Parámetros requeridos en el cuerpo de la solicitud:

PARÁMETRO	VALOR	ORIGEN
grant_type	authorization_code	Por defecto
code	*"TU_CODIGO_AUTORIZACION"*	Codigo
redirect_uri	https://miapp.entidad.gob.pe/auth/callback	Configuración
client_id	TU_CLIENT_ID	Configuración
client_secret	TU_CLIENT_SECRET	Configuración

Tabla 7: parámetros para el endpoint /token

Consideraciones técnicas.

- ✓ Todos los parámetros incluidos en la solicitud POST deben estar codificados usando URL encoding (application/x-www-form-urlencoded), conforme al estándar.
- ✓ Correspondencia exacta del valor del parámetro establecido en "***redirect_uri***", el valor de este parámetro cuando fue enviado en la solicitud al endpoint "***/token***":
  - Debe coincidir exactamente (carácter por carácter) con el utilizado previamente al construir la URL de autenticación.
  - Debe ser idéntico al valor registrado ante RENIEC.
  - Cualquier diferencia en el protocolo (https), host, ruta o codificación puede resultar en un rechazo.
- ✓ Si el valor del parámetro "***code***" es inválido, expirado o ha sido reutilizado, se retornará un error tipo:

```
{
    "error": "invalid_code",
    "error_description": "No exist code."
}
```

Código Fuente 19: respuesta de error del parámetro code

- ✓ Por último, procedemos al intercambio del parámetro "***code***" por un token del tipo JWT (Jasen Web Token).

```
IF code IS VALID:
    headers = {
        "Content-Type": "application/x-www-form-urlencoded"
    }

    body = {
        "grant_type": "authorization_code",
        "code": code,
        "redirect_uri": redirect_uri,  //archivo de configuración
        "client_id": client_id,  //archivo de configuración
        "client_secret": client_secret,  //archivo de configuración
    }

    response = http.post(headers, body)

    IF response.status == 200:
        access_token = response.body["access_token"]
        token_type = response.body["token_type"]
        expires_in = response.body["expires_in"]
        id_token = response.body["id_token"]
    ELSE:
        LOG_ERROR("Fallo en intercambio de token: " +
response.body["error"])
        redirigirAError()
```

Código Fuente 20: Intercambio del valor del parámetro code por un valor token

- ✓ Es fundamental tener en cuenta que el token de identificación "***id_token***" recibido como parte de la respuesta al endpoint "***/token***" es un JSON Web Token (JWT). Este token contiene información estructurada, codificada y firmada digitalmente, que representa la identidad autenticada del ciudadano. Su correcta decodificación y validación es esencial para asegurar la confiabilidad del proceso de autenticación antes de usar la información en la lógica de negocio.

Recomendaciones de implementación

- ✓ Implementar registro de logs detallado para almacenar tanto el código recibido como los intentos de intercambio.
- ✓ Si se detecta error, invalidar la sesión del usuario y redirigir a una ruta segura, ofreciendo reiniciar el proceso de autenticación.
- ✓ Asegurar que el "***code***" no se reprocese más de una vez por sesión.

Respuesta Esperada del Endpoint "***/token***"

Si la solicitud al endpoint "***/token***" ha sido realizada correctamente (parámetros válidos, autenticación exitosa y código no expirado), el servicio ID Perú retornará una respuesta en formato JSON que contiene los tokens necesarios para continuar con el proceso de identificación e integración de datos del usuario autenticado.

```
{
  "access_token": "dFRgblR8u…",
  "token_type": "bearer",
  "expires_in": 604800,
  "id_token": "eyJhbGciOiJSUzI1Ni…JfWxvAWt_yOMrq…ULATiN2u3Yuo4dbPw…"
}
```

Código Fuente 21: Json con el token JWT de ejemplo

Descripción de los campos

CAMPO	DESCRIPCIÓN
access_token	Token de acceso que permite consumir el endpoint "***/userinfo***" para obtener los datos del ciudadano autenticado. Debe ser manejado de forma segura.
token_type	Tipo de token retornado. Para ID Perú, este valor es siempre bearer.
expires_in	Tiempo de vida del access_token, expresado en segundos.
id_token	Token JWT que contiene información codificada sobre la autenticación y la identidad validada. Se puede validar y decodificar localmente mediante las claves públicas del proveedor endpoint "***/certs***".

Tabla 8: Descripción de los campos del Token JWT

Consideraciones de seguridad.

- ✓ Los valores en los parámetros "***access_token***" y "***id_token***" deben ser almacenados y transmitidos de forma segura, siguiendo las buenas prácticas de seguridad (no exponer en URLs, no persistir sin cifrado, etc.).
- ✓ El parámetro "***access_token***" no debe ser compartido ni reutilizado fuera del contexto de sesión autenticada del usuario.
- ✓ Se recomienda validar la firma del "***id_token***" utilizando las claves públicas disponibles en el endpoint establecido en el parámetro "***auth_provider_keys_uri***", que se ubica en el archivo de configuración.

3. LECTURA Y DECODIFICACIÓN DEL TOKEN JWT.

Como parte de la respuesta recibida desde el endpoint "***/token***", se obtiene un parámetro denominado "***id_token***", el cual contiene un JSON Web Token (JWT). Este token representa un conjunto de afirmaciones "***claims***" sobre la identidad autenticada, y es generado, firmado y emitido por la plataforma ID Perú.

Para continuar con el proceso de identificación, es necesario leer y decodificar el contenido del token JWT, particularmente la sección conocida como "***payload***", que incluye los datos de identidad del ciudadano identificado.

Estructura de un token JWT

Un token JWT está compuesto por tres secciones codificadas en Base64URL, separadas por puntos (.), en el siguiente orden:

```
[HEADER].[PAYLOAD].[SIGNATURE]
```

Código Fuente 22: estructura de un token JWT

SECCIÓN	DESCRIPCIÓN
Header	Metadatos del algoritmo y tipo de token (usualmente RS256, JWT).
Payload	Contiene los datos o claims sobre el sujeto autenticado.

Signature	Firma digital generada con la clave privada del proveedor.

Tabla 9: descripción de la estructura de un token JWT

Objetivo de esta sección

En esta etapa, solo se decodificará el segmento payload, con el propósito de obtener los datos del usuario autenticado por ID Perú, como, por ejemplo: DNI y nombres, etc.

Ejemplo visual de un Token JWT

```
eyJhbGciOiJSUzI1NiIsInR5cCI6IkpXVCJ9. // HEADER
eyJzdWIiOiIxMjM0NTY3ODkwIiwibmF2IucGUifQ. // PAYLOAD
G1EOYsw8yyu2fWJ7-EXAMPLE-BVJFnftK9zNkT7ADbk // SIGNATURE
```

Tabla 10: token JWT

Proceso de decodificación del segmento denominado “payload” del JWT.

Luego de fragmentar el valor del parámetro “***id_token***” y extraer su sección central “***payload***”, se procede a decodificar dicho segmento desde Base64URL. El resultado será una cadena JSON con los “***claims***” de identidad y seguridad firmados por el servicio ID Perú.

```
FUNC decodificarPayloadIdToken(id_token: STRING) RETURNS JSON:
    segmentos = id_token.split(".")
    payloadCodificado = segmentos[1]

    // Base64URL decode (sin padding)
    payloadJSON = base64UrlDecode(payloadCodificado)

    datos = parseJSON(payloadJSON)
    RETURN datos
```

Código Fuente 23:Decodificar payload

Advertencias técnicas

- ✓ No modificar ni alterar el token antes de su lectura.
- ✓ El “***id_token***” puede ser verificado criptográficamente con la clave pública disponible en la siguiente dirección de URL “***endpoint***”:

```
https://idaas.reniec.gob.pe/service/certs
```

Código Fuente 24: URL de claves públicas para validar token JWT

Esta decodificación no requiere claves públicas, ya que no se está validando la firma en esta etapa, sino únicamente accediendo al contenido.

```
{
 tokenType='bearer',
 accessToken='dFRgblR8uG4y…',
 expiresIn=604800,
 idToken='eyJhbGciO…4OWE2MjFlOTQ…qQd1IbiV_4QcyCUwtL…',
 refreshToken='null',
 payload ='{
        "iss":"https:\/\/idaas.reniec.gob.pe\/service",
         "sub":"98726b…",
        "aud":["FhPezs7CNJvq0…"],
        "iat":1747697099,
        "exp":1747700699,
```

```
        "c_hash":"TLGlhAuVmA…",
        "at_hash":"YKb5ZEMq6S…",
        "acr":"face_mobile"
       }',

       idToken= {
        acr='face_mobile',
        aud=[FhPezs7CN…],
        sub='98726b2b…',
        doc='null',
        firstName='null',
        email='null',
        emailVerified=null,
        phoneNumber='null',
        phoneNumberVerified=null,
        exp=1747700699,
        iat=1747697099,
        iss='https://idaas.reniec.gob.pe/service',
        nonce='null',
        athash='YKb5ZEM…',
        chash='TLGlhA…'
       }
}
```

Código Fuente 25: Ejemplo de segmento Payload decodificado

Para obtener información completa del usuario, se procederá a realizar la consulta a la dirección URL del tipo endpoint "***/userinfo***" usando el valor en establecido en el parámetro "***access_token***". Con el que accederemos eficazmente a los datos del usuario.

4. CONSULTA DE LOS DATOS DE IDENTIFICACIÓN.

Una vez obtenido un valor correcto establecido en el parámetro "***access_token***" y habiendo verificado correctamente la autenticación del ciudadano mediante los pasos anteriores, el sistema cliente puede realizar la consulta de los datos de identificación del ciudadano identificado a través de la dirección URL del tipo endpoint "***/userinfo***" expuesto por RENIEC.

Este servicio devuelve los atributos personales autorizados para el uso por parte de la entidad integradora, de acuerdo al convenio interinstitucional vigente.

Especificaciones para la solicitud a la dirección URL del tipo endpoint "***/userinfo***".

- ✓ Método HTTP: POST
- ✓ URL: https://idaas.reniec.gob.pe/service/userinfo
- ✓ Content-Type: application/x-www-form-urlencoded
- ✓ Parámetros requeridos en el cuerpo de la solicitud:

PARÁMETRO	VALOR
Authorization	Bearer dFRgblR8…

Tabla 11: Parámetros para el endpoint /userinfo

Respuesta en formato json con el siguiente contenido

```
{
"sub": "98726b2b…",
"doc": "12345678",
"first_name": "Homer"
```

```
}
```

Código Fuente 26: Json de respuesta del endpoint /userinfo

Descripción de los campos retornados:

CAMPO	DESCRIPCIÓN
sub	Identificador único y encriptado asignado por RENIEC. Representa al ciudadano autenticado.
doc	Número del Documento Nacional de Identidad (DNI) del ciudadano.
first_name	Primer nombre del ciudadano autenticado.

Tabla 12: Descripción de parámetros de respuesta de /userinfo

Consideraciones importantes.

- ✓ La dirección URL del tipo endpoint "***/userinfo***" puede ser consultado mientras el valor establecido en el parámetro "***access_token***" sea válido.
- ✓ Solo se puede consultar la identidad autenticada; no está permitido ni es posible recuperar información de otro ciudadano durante la misma sesión.
- ✓ Si la respuesta contiene campos null, incompletos o inesperados:
  - Verifique si los scopes solicitados en la autenticación cubren dichos atributos.
  - Valide si la identidad contiene la información esperada en RENIEC.
  - Contacte al área de soporte técnico de RENIEC en caso de inconsistencias.

Muestra de ejemplo en seudocódigo.

```
FUNC consultarDatosIdentidad(access_token: STRING):
    headers = {
        "Authorization": "Bearer " + access_token,
        "Content-Type": "application/x-www-form-urlencoded"
    }

    response = config.userinfo_uri  //archivo de configuración

    IF response.status == 200:
        datos = parseJson(response.body)
        LOG("Identificación exitosa del ciudadano")
        REDIRIGIR A vista.logicaDeNegocio
    ELSE:
        LOG_ERROR("Error en la consulta a /userinfo: " + response.status)
        REDIRIGIR A vista.controlada
```

Código Fuente 27: consulta al endpoint /userinfo

10. LÓGICA DE NEGOCIO INTERNA.

Una vez completado exitosamente el proceso de autenticación e identificación del ciudadano mediante la plataforma ID Perú, y habiéndose obtenido los datos personales a través de la dirección URL del tipo endpoint *"**/userinfo***", el flujo técnico del servicio de ID Perú proporcionado por RENIEC puede considerarse concluido satisfactoriamente.

A partir de este punto, la aplicación integradora deberá encargarse de procesar y utilizar los datos recibidos conforme a su propia lógica de negocio interna.

1. Responsabilidad de la Aplicación Integradora.

Con la información del ciudadano ya validada e integrada, el sistema cliente (entidad pública o privada) debe tomar decisiones sobre cómo usar esta identidad autenticada. Algunas acciones comunes incluyen:

Casos típicos de uso:

ESCENARIO	EJEMPLO
Inicio de sesión seguro	Registrar al usuario en sesión como autenticado (session["usuario"] = datos) y redirigirlo al sistema.
Creación o asociación de cuenta	Si el usuario no existe en el sistema local, crear un nuevo registro con los datos autenticados.
Validación previa a trámites	Usar los valores de los parámetros "***doc***" o "***sub***" para validar si el ciudadano está autorizado a iniciar un trámite o proceso.
Acceso personalizado	Cargar datos asociados a la identidad del ciudadano previamente almacenados (beneficios, servicios, historiales, etc.).

Tabla 13:casos de uso para el servicio de RENIEC

2. Consideraciones técnicas

- ✓ La lógica de negocio posterior al proceso de identificación no depende de RENIEC y debe ser definida íntegramente por la institución integradora.
- ✓ La aplicación puede decidir almacenar temporalmente los datos del usuario, asociarlos con un sistema de gestión interna (CRM, ERP, etc.), o integrarlos con otros servicios propios.
- ✓ Toda la gestión post-identificación debe respetar los principios de seguridad, confidencialidad y trazabilidad, especialmente si los datos serán persistidos o auditados.

3. Buenas prácticas recomendadas.

- ✓ Asociar el sub recibido como identificador interno de autenticación, ya que es único, estable y cifrado por RENIEC.
- ✓ Validar si el usuario ya existe en el sistema y, si no, permitir un flujo de registro semiautomático con los datos recibidos.
- ✓ Establecer una política de expiración de sesión basada en la validez del valor establecido en el parámetro "***access_token***" o en los tiempos definidos por la institución.

11. RESUMEN DEL FLUJO DE AUTENTICACIÓN.

1. Inicio de la autenticación (Usuario → Sitio Web)

- El usuario hace clic en el botón de autenticación.
- La aplicación integradora redirige al navegador del usuario a la dirección URL del tipo endpoint "***/auth***".
- En esa redirección se incluyen los siguientes parámetros:
  - client_id: Identificador único proporcionado por RENIEC.
  - response_type: Se solicita un código de autorización "***code***".
  - scope: Qué datos se desean mínimo "***openid***".
  - redirect_uri: URL a la que se enviará la respuesta registrada en RENIEC.
  - state: Identificador de sesión (evita ataques CSRF).

2. Identificación en IDPERÚ (Servicio ID Perú)

- El usuario visualiza la pantalla de identificación de ID PERÚ y se autentica con:
  - Credenciales solicitadas por el servicio ID Perú
  - Autenticación biométrica.
  - Documento de Identidad electrónico (DNIe)

3. Autenticación exitosa (IDPERÚ → Sitio Web)

- IDPERÚ valida las credenciales.
- Si son correctas, redirige a la dirección URL establecida en el parámetro "***redirect_uri***" registrado previamente en RENIEC.
- En esta redirección se incluye en la URL:
  - code: El código de autorización (de un solo uso).
  - state: El mismo valor enviado al principio para verificar integridad.

4. Canje de código por token (Sitio Web → IDPERÚ)

- La aplicación integradora realiza una solicitud POST a la dirección URL del tipo endpoint "***/token***".
- Se envían los siguientes datos:
  - client_id
  - client_secret
  - redirect_uri
  - grant_type=authorization_code
  - code: El recibido previamente.

5. Recepción del Token

- El Servicio ID PERÚ responde con un JSON que incluye el:
  - access_token: Token de acceso
  - id_token: Token JWT con datos
  - expires_in: tiempo de valides del token

- token_type: tipo de token “Bearer”

6. Solicitud de datos del usuario (Sitio Web → IDPERÚ)

- La aplicación integradora hace un POST a la dirección URL del tipo endpoint “***/userinfo***” con:
  - Header = Authorization: Bearer {access_token}

7. Datos del usuario autenticado

- IDPERÚ devuelve un JSON con la información del usuario:
- doc, nombre y sub.

8. Uso de datos en el sitio integrado

- La aplicación integradora aplica la lógica de negocio con a la información recibida.

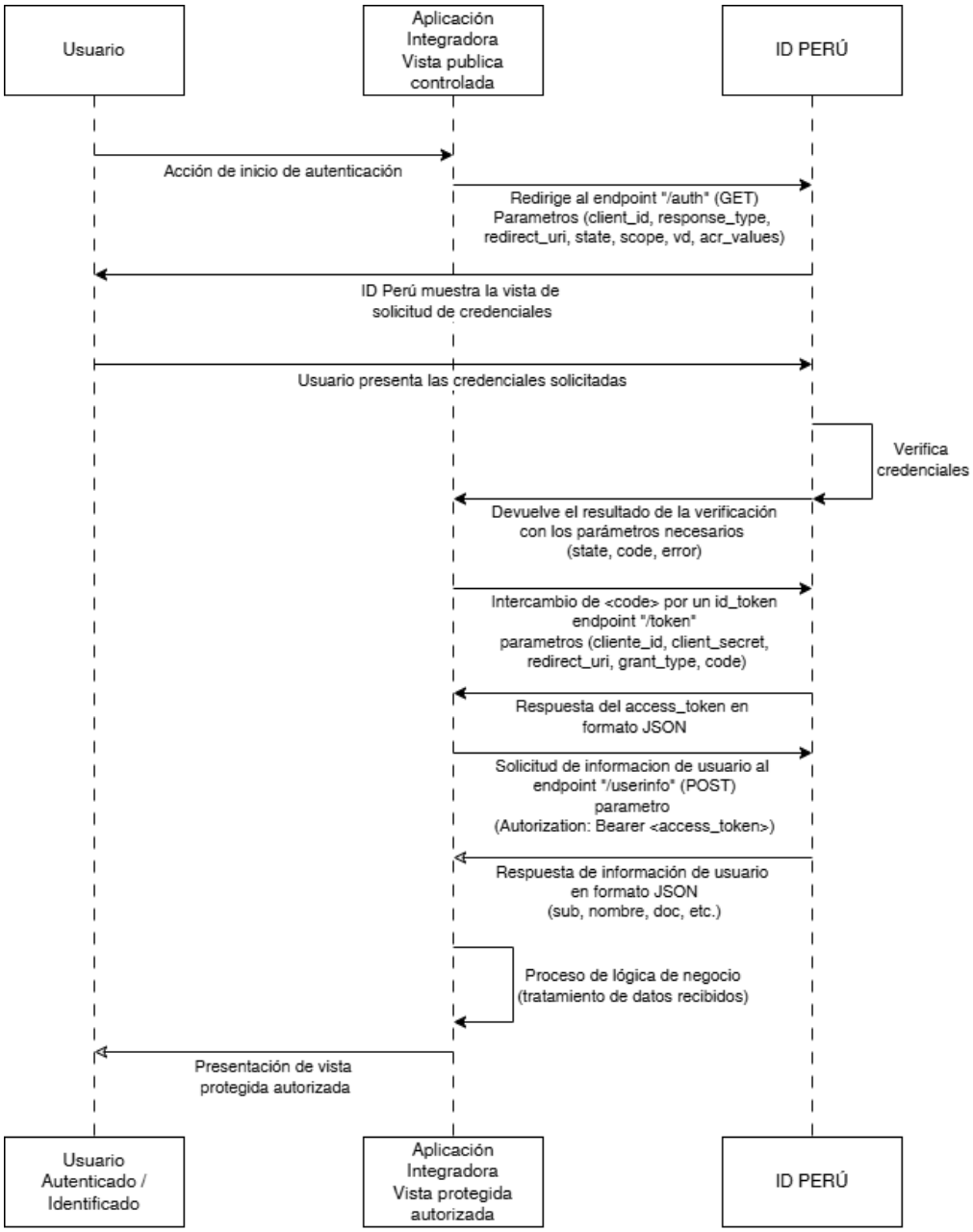

Imagen 2:Flujo de servicio ID Perú

12. CONCLUSIONES

El flujo de identificación de ciudadano implementado por ID Perú proporciona una forma estandarizada, segura y escalable de autenticar usuarios con nacionalidad peruana (número de DNI indispensable) en aplicaciones gubernamentales o privadas. Entre sus ventajas están:

- ✓ Usa estándares ampliamente soportados.
- ✓ Evita el manejo de credenciales en el sitio web.
- ✓ Facilita la autenticación federada con biometría.
- ✓ Permite asociar los datos del ciudadano con sistemas internos fácilmente.

Es importante que los desarrolladores manejen correctamente:

- ✓ El "***state***" para evitar CSRF.
- ✓ El intercambio seguro del "***code***".
- ✓ La validación del "***id_token***" y "***access_token***".

13.RECOMENDACIONES

- ✓ Separación de responsabilidades: Divide claramente las responsabilidades entre las capas: autenticación, validación del token, consulta de datos y lógica de negocio. Esto mejora el mantenimiento y la seguridad.
- ✓ Gestión segura de credenciales: Nunca almacenes el "***client_secret***" en texto plano ni lo expongas en el cliente. Usa un almacén seguro (como variables de entorno o servicios de secretos).
- ✓ Validación estricta del parámetro "***state***": Siempre valida que el "***state***" recibido coincida con el que fue generado y almacenado en la sesión HTTP activa antes de intercambiar el valor del parámetro "***code***".
- ✓ Validación del valor establecido en el paramtro "***id_token***" en producción: Valida siempre la firma digital, la expiración "***exp***" y la audiencia "***aud***" del valor en el parámetro "***id_token***" usando las claves públicas expuestas por RENIEC.
- ✓ Protección contra ataques de repetición: Asegúrate de invalidar el valor establecido en el parámetro "***code***" una vez que haya sido usado. Nunca reutilices un token para múltiples sesiones.
- ✓ Auditoría y trazabilidad: Registra cada paso crítico (solicitud, redirección, obtención de tokens, errores) sin almacenar información sensible como tokens completos o datos personales.
- ✓ Revisiones del convenio: Asegúrate de que los valores establecidos en el parámetro "***scopes***" y atributos requeridos estén correctamente habilitados en el convenio con RENIEC.
- ✓ Monitoreo de disponibilidad: Implementa alertas ante fallos en las direcciones URL del tipo endpoints "***/token***" y "***/userinfo***" para detectar posibles caídas o cambios de comportamiento.

14. LENGUAJES DE PROGRAMACIÓN RECOMENDADOS

A continuación, los lenguajes y entornos más recomendados para implementar este servicio, con soporte sólido para JWT y HTTPS:

LENGUAJE	POR QUÉ SE RECOMIENDA	BIBLIOTECAS COMUNES
Java (Spring Boot)	Ideal para entornos corporativos, soporte nativo para OAuth2 y JWT.	spring-security, Nimbus, JSON y JWT

Node.js	Muy usado en integraciones rápidas y aplicaciones web SPA.	passport, jsonwebtoken, axios
Python	Rápido para prototipos.	requests, oauthlib, PyJWT
.NET (C#)	Robusto para apps empresariales, integra bien con servicios.	Microsoft.AspNetCore.Authentication.JwtBearer
Go (Golang)	Ligero y seguro para servicios backend.	golang.org/x/oauth2, jwt-go

Tabla 14: Leguajes de programación recomendados

Recomendación adicional: en cualquier lenguaje, asegúrate de usar bibliotecas activamente mantenidas y con soporte.

15. GLOSARIO DE TÉRMINOS

TÉRMINO	DEFINICIÓN
OpenID Connect	Extensión de OAuth2 utilizada específicamente para autenticación de usuarios.
client_id	Identificador único asignado por RENIEC a cada entidad integradora.
client_secret	Clave secreta asociada al "***client_id***" para autenticar solicitudes.
access_token	Token temporal utilizado para acceder al endpoint "***/userinfo***".
id_token	Token en formato JWT que contiene información sobre la identidad autenticada.
state	Valor único generado por el cliente para correlacionar solicitud y respuesta.
redirect_uri	URL de retorno registrada donde ID Perú redirige tras la autenticación.
JWT	JSON Web Token, formato estándar para representación de claims seguros.
scope	Permisos o atributos solicitados al usuario durante la autenticación.
sub	Identificador único encriptado del ciudadano autenticado.
userinfo	Endpoint protegido que retorna los datos del ciudadano autenticado.
aud	Audiencia. Representa a quién está destinado el token, usualmente el client_id.
exp	Expiración del token en formato epoch.
iat	Fecha de emisión del token.
acr	Nivel de autenticación utilizado (ej. face_mobile, two_factor, etc.).
token_type	Tipo de token retornado, normalmente 'bearer'.
refresh_token	Token opcional usado para obtener nuevos access_tokens sin reautenticación.
authorization_code	Código temporal que representa una sesión de autenticación exitosa.

Tabla 15: Glosario de términos

16. HISTORIAL DE MODIFICACIONES

VERSIÓN	ESTADO	FECHA
v1.0	Aprobado	12/06/2024
v2.0	Revisión	27/05/2025
v2.1	Revisión	16/06/2025
V2.1	Aprobado	18/07/2025

Tabla 16: Historial de versiones