Integración con Active Directory mediante SCIM

Descripción General

El sistema debe implementar una integración completa con Active Directory (AD) utilizando el protocolo **SCIM 2.0** (System for Cross-domain Identity Management) para la sincronización unidireccional de usuarios, grupos y roles desde el AD hacia la plataforma. La autenticación se realizará mediante **SAML 2.0** (Single Sign-On), y el sistema debe soportar múltiples clientes (multitenant) cada uno con su propio Active Directory.

Características Principales

- **Sincronización unidireccional**: AD → Plataforma (AD es la fuente de verdad)
- **Autenticación**: SAML 2.0 (protocolo no negociable)
- **Aprovisionamiento**: SCIM 2.0 con sincronización en tiempo real
- **Duración de sesión**: 4 horas
- **Invalidación proactiva**: Sesiones activas se invalidan ante cambios críticos de permisos
- **Arquitectura multitenant**: Soporte para múltiples clientes con diferentes ADs
- **Coexistencia**: Usuarios gestionados vía AD y usuarios locales en la misma plataforma
- **Mapeo de roles**: Replicación directa - el cliente es responsable de usar exactamente los nombres de rol de la plataforma

---

1. Arquitectura de Integración

1.1 Componentes del Sistema

Componente 1: Endpoint SCIM

- La plataforma expone un endpoint SCIM único que recibe eventos de sincronización
- El endpoint diferencia entre clientes mediante un identificador de cliente (tenant ID)
- Ruta sugerida: `https://plataforma.com/scim/v2/{tenantId}/Users`

Componente 2: SCIM Client (Active Directory)

- El AD de cada cliente actúa como SCIM client
- Envía eventos de sincronización al endpoint SCIM de la plataforma
- Se autentica mediante Bearer token

Componente 3: Identity Provider (IdP) - SAML 2.0

- El AD actúa también como Identity Provider para autenticación SSO
- Emite assertions SAML con atributos del usuario (nombre, email, grupos, roles)

Componente 4: Catálogo de Roles de la Plataforma

- Lista de roles definidos en la plataforma que el cliente debe replicar exactamente en su AD
- Disponible para exportación en interfaz de administración
- El cliente es responsable de crear grupos/roles en AD con estos nombres exactos (case-sensitive)

Componente 5: Gestor de Sesiones

- Gestiona tokens de sesión con duración de 4 horas
- Implementa invalidación proactiva de sesiones ante cambios críticos

Componente 6: Motor de Segmentación Local

- Gestiona prefijos y niveles de segmentación exclusivamente en la plataforma
- No se sincroniza con AD (diseño preparado para futura extensión)

---

2. Sincronización mediante SCIM

2.1 Sincronización Inicial (Importación Masiva)

Cuando un cliente activa la integración con AD por primera vez:

Paso 1: Provisión de Catálogo de Roles

- Cadena proporciona al cliente un listado exportable de todos los roles disponibles en la plataforma
- El cliente revisa el catálogo y comprueba que su AD tiene grupos/roles con exactamente los mismos nombres
- **Responsabilidad del cliente**: Crear en su AD los grupos/roles faltantes con los nombres exactos de la plataforma (case-sensitive)

Paso 2: Configuración de Filtros

- El cliente define qué grupos o roles del AD se sincronizarán con la plataforma
- No se importan TODOS los usuarios del AD, solo los que pertenecen a los grupos/roles especificados
- Ejemplo: "Sincronizar solo usuarios de los grupos: Administrador, Gestor, Usuario"
- **Nota**: Los nombres de grupos DEBEN coincidir exactamente con los roles del catálogo

Paso 3: Autenticación del SCIM Client

- El AD (SCIM client) se autentica contra el endpoint SCIM mediante Bearer token
- El token identifica al cliente (tenant) y valida permisos de escritura

Paso 4: Importación Masiva

- El AD envía solicitudes SCIM POST para crear cada usuario
- Atributos sincronizados: nombre completo, email, grupos, roles
- Cada grupo/rol que llega en el payload SCIM se valida contra el catálogo de roles de la plataforma

Paso 5: Validación de Roles y Creación de Usuarios

- Sistema busca cada grupo/rol recibido en el catálogo de roles de la plataforma
- **Coincidencia exacta y case-sensitive**: `Administrador` ≠ `administrador`
- Si el grupo/rol existe en el catálogo: se asigna al usuario
- Si el grupo/rol NO existe en el catálogo: se **omite silenciosamente** (no otorga permiso)
- Los usuarios se crean en la base de datos con los roles válidos asignados

Paso 6: Creación de Usuarios

- Los usuarios se crean en la base de datos de la plataforma con estado "activo"
- Se asocian al cliente (tenant) correspondiente
- Se registra la fuente de autenticación: "AD" (vs "Local")

2.2 Sincronización en Tiempo Real

Después de la sincronización inicial, el sistema opera en modo de sincronización en tiempo real:

Mecanismo: Webhooks/Notificaciones

- Cada cambio en el AD genera un evento SCIM que se envía inmediatamente a la plataforma
- No se implementa sincronización batch peridica en el alcance inicial (queda en backlog)

Eventos Soportados:

Evento en AD	Operación SCIM	Acción en Plataforma
Creación de usuario	POST /Users	Crear usuario localmente (validar roles contra catálogo)
Modificación de atributos (nombre, email)	PATCH /Users/{id}	Actualizar atributos localmente
Cambio de membresía de grupos	PATCH /Users/{id}	Actualizar grupos y validar contra catálogo de roles

Modificación de roles asignados	PATCH /Users/{id}	Validar y actualizar roles contra catálogo
Deshabilitación de usuario	PATCH /Users/{id} con `active: false`	Marcar usuario como inactivo + invalidar sesiones
Habilitación de usuario	PATCH /Users/{id} con `active: true`	Reactivar usuario
Eliminación de usuario	DELETE /Users/{id}	Soft delete (marcar como eliminado) + invalidar sesiones

Flujo de Procesamiento:

1. AD detecta cambio en usuario
2. AD (SCIM client) envía evento SCIM al endpoint de la plataforma
3. Plataforma valida autenticación (Bearer token)
4. Plataforma valida idempotencia (evita duplicación)
5. Plataforma procesa evento según tipo de operación
6. **Plataforma valida cada grupo/rol contra catálogo** (coincidencia exacta, case-sensitive)
7. Plataforma actualiza base de datos (asignando solo roles válidos)
8. Si es cambio crítico → Invalida sesiones activas
9. Plataforma registra evento en logs de auditoría
10. Plataforma responde con código HTTP 200 (éxito) o 5xx (error transitorio)

2.3 Operaciones SCIM Soportadas

El endpoint SCIM debe soportar el siguiente subconjunto de operaciones SCIM 2.0:

Operaciones Obligatorias:

- **POST /Users**: Crear usuario
- **PATCH /Users/{id}**: Actualizar usuario (atributos, grupos, roles, estado activo/inactivo)
- **DELETE /Users/{id}**: Eliminar usuario (soft delete)

Operaciones NO Requeridas (Alcance Inicial):

- **GET /Users/{id}**: Consultar usuario individual (no requerido inicialmente)
- **GET /Users?filter=...**: Búsquedas y filtrado (queda en backlog)
- **PUT /Users/{id}**: Reemplazo completo de usuario (se usa PATCH en su lugar)

Idempotencia:

- Todas las operaciones SCIM deben ser idempotentes
- Si se recibe el mismo evento varias veces (ej: por reintentos), el resultado debe ser el mismo
- Ejemplo: PATCH para desactivar un usuario ya inactivo no debe generar error

---

3. Autenticación mediante SAML 2.0

3.1 Flujo de Autenticación SSO

Paso 1: Usuario accede a la plataforma

- Usuario ingresa a la URL de la plataforma
- Sistema presenta login con selección de cliente/empresa

Paso 2: Selección de Cliente

- Usuario selecciona el cliente (tenant) con el que desea autenticarse
- Sistema obtiene la configuración del cliente desde la base de datos

Paso 3: Determinación del Método de Autenticación

- Sistema verifica si el cliente tiene integración con AD activada
- Si NO tiene AD: redirige a login local con usuario/contraseña
- Si SÍ tiene AD: continúa con flujo SAML

Paso 4: Redirección al Identity Provider (IdP)

- Sistema genera una solicitud SAML (AuthnRequest)
- Redirige al usuario al IdP del AD del cliente

Paso 5: Autenticación en Active Directory

- Usuario se autentica en el AD
- AD valida las credenciales

Paso 6: Generación de Assertion SAML

- AD genera una assertion SAML firmada con los atributos del usuario:
  - `nameID`: identificador único del usuario (email)
  - `givenName`: nombre
  - `surname`: apellido
  - `email`: correo electrónico
  - `groups`: lista de grupos del AD a los que pertenece
  - `roles`: lista de roles del AD asignados

Paso 7: Validación de Assertion SAML

- Plataforma valida la firma digital de la assertion
- Plataforma verifica que la assertion no haya expirado
- Plataforma extrae los atributos del usuario

Paso 8: Validación de Roles contra Catálogo

- Plataforma extrae los grupos/roles de la assertion SAML
- **Búsqueda en catálogo**: Cada grupo/rol se busca en el catálogo de roles de la plataforma
- **Coincidencia exacta y case-sensitive**: El nombre debe coincidir exactamente
- **Regla de validación**: Si un grupo/rol del AD no existe en el catálogo, se omite silenciosamente (no otorga ningún permiso)
- Solo los grupos/roles que coincidan exactamente se asignan al usuario

Paso 9: Generación de Token de Sesión

- Plataforma genera un token JWT con duración de **4 horas**
- Token incluye: ID de usuario, ID de cliente, roles asignados (solo los válidos), timestamp de expiración
- Token se firma criptográficamente

Paso 10: Acceso a la Plataforma

- Usuario es redirigido a la plataforma con sesión activa
- Puede acceder a módulos según sus roles válidos
- Información visible dentro de módulos se filtra según segmentación local (prefijos)

3.2 Validación de Roles y Acceso a Módulos

- **Primera validación (Roles)**: ¿El usuario tiene un rol válido (que coincida exactamente en el catálogo) para acceder al módulo?
  - Si NO: denegar acceso al módulo
  - Si SÍ: continuar
- **Segunda validación (Segmentación)**: ¿El cliente tiene segmentación activada?
  - Si NO: usuario ve toda la información del módulo
  - Si SÍ: usuario ve solo información de los prefijos asignados localmente

---

4. Invalidación Proactiva de Sesiones

4.1 Cambios Críticos de Permisos

El sistema debe invalidar inmediatamente las sesiones activas cuando detecta los siguientes cambios críticos en el AD:

Cambio Crítico	Evento SCIM	Acción
Deshabilitación de usuario	PATCH con `active: false`	Invalidar todas las sesiones del usuario
Eliminación de usuario	DELETE /Users/{id}	Invalidar todas las sesiones del usuario
Remoción de un rol asignado	PATCH con modificación de `roles`	Invalidar todas las sesiones del usuario
Remoción de membresía de grupo	PATCH con modificación de `groups`	Invalidar todas las sesiones del usuario

4.2 Mecanismo de Invalidación

Paso 1: Detección de Cambio Crítico

- Al procesar un evento SCIM, el sistema verifica si se trata de un cambio crítico
- Comparación: estado/roles/grupos anteriores vs nuevos

Paso 2: Invalidación de Token de Sesión

- Sistema marca el token JWT del usuario como inválido en una lista negra (blacklist)
- Alternativa: Sistema elimina el token de la sesión activa en caché

Paso 3: Notificación al Usuario

- En el siguiente request del usuario, el sistema detecta que su sesión fue invalidada
- Sistema muestra mensaje: "Su sesión ha sido cerrada por cambios en sus permisos. Por favor inicie sesión nuevamente."
- Usuario es redirigido al login

Paso 4: Reautenticación Obligatoria

- Usuario debe volver a autenticarse mediante SAML 2.0
- Sistema obtiene los permisos actualizados del AD
- Se genera un nuevo token con los permisos correctos

4.3 Ventana de Exposición Reducida

- **Objetivo**: Reducir la ventana de exposición de 8 horas (JIT) a **minutos**
- **Escenario crítico mitigado**: Si un empleado es despedido y deshabilitado en AD, su sesión activa se cierra en menos de 1 minuto (tiempo de propagación del evento SCIM)

---

5. Autenticación del SCIM Client

5.1 Método de Autenticación: Bearer Token

Generación de Token:

- La plataforma genera un Bearer token único para cada cliente (tenant)
- Token se entrega al administrador del cliente para configurar el SCIM client en su AD
- Formato: JWT firmado o token alfanumérico seguro (mínimo 32 caracteres)

Uso del Token:

- El AD (SCIM client) incluye el token en cada request SCIM:

```
Authorization: Bearer <token>
```

- La plataforma valida el token en cada request

- Si el token es inválido o expirado, la plataforma responde con HTTP 401 Unauthorized

Validación:

- Token identifica al cliente (tenant)
- Token valida permisos de escritura en el endpoint SCIM
- Token tiene fecha de expiración configurable

5.2 Rotación de Credenciales

Política de Rotación:

- Los Bearer tokens deben rotarse periódicamente (recomendado: cada 90 días)
- La plataforma permite tener dos tokens activos simultáneamente durante el período de transición
- Esto permite al cliente actualizar la configuración en AD sin interrumpir la sincronización

Proceso de Rotación:

1. Administrador de la plataforma genera un nuevo token para el cliente
2. Ambos tokens (viejo y nuevo) son válidos durante 7 días
3. Administrador del cliente actualiza la configuración del SCIM client en AD con el nuevo token
4. Después de 7 días, el token viejo expira automáticamente
5. Solo el nuevo token permanece activo

Alertas:

- Sistema envía alertas al administrador cuando un token está próximo a expirar (15 días antes)
- Sistema alerta si detecta uso de un token expirado (intento de sincronización fallido)

---

6. Estrategia de Mapeo de Roles (Replicación Directa)

6.1 Principio: Responsabilidad del Cliente

No existe tabla de homologación en la plataforma.

El cliente es responsable de configurar en su Active Directory grupos/roles con exactamente los mismos nombres que aparecen en el catálogo de roles de la plataforma.

Beneficios de este enfoque:

- Elimina complejidad de desarrollo (sin interfaz de administración para mapeos)
- Acelera el go-live (implementación más rápida)
- Evita ambigüedades y errores de mapeo
- Responsabilidad clara: cliente replica exactamente los nombres

6.2 Catálogo de Roles de la Plataforma

La plataforma expone un catálogo de roles disponible en:

- **Interfaz de administración**: Vista exportable de todos los roles
- **Documentación**: Listado en formato PDF/CSV
- **API**: Endpoint GET /admin/roles para consulta programática

Ejemplo de Catálogo:

```
Administrador
Auditor
Analista
Gestor
Supervisor
Usuario
```

6.3 Responsabilidades Compartidas

Responsabilidades de la Plataforma:

- Mantener y actualizar el catálogo de roles
- Exponer el catálogo para que el cliente lo descargue/consulte
- Validar cada grupo/rol contra el catálogo durante sincronización SCIM
- Validar cada grupo/rol contra el catálogo durante autenticación SAML
- Omitir silenciosamente roles que no existan en el catálogo
- Documentar claramente el catálogo de roles

Responsabilidades del Cliente:

- Descargar el catálogo de roles de la plataforma
- Crear en su AD grupos/roles con exactamente los mismos nombres
- Usar coincidencia exacta (case-sensitive): `Administrador` ≠ `administrador`
- Asignar usuarios a estos grupos en su AD
- Validar que los nombres coincidan antes de configurar la sincronización SCIM
- Notificar a la plataforma si hay inconsistencias o cambios en el catálogo

6.4 Manejo de Inconsistencias

Escenario 1: Grupo en AD que no existe en catálogo

- Ejemplo: Usuario tiene grupo `Admin_TI` en AD, pero el catálogo de la plataforma tiene `Administrador`
- **Acción**: Se omite silenciosamente; usuario no obtiene ese permiso
- **Solución**: Cliente debe renombrar el grupo en AD a `Administrador` o contactar a la plataforma para agregar `Admin_TI` al catálogo

Escenario 2: Diferencia en mayúsculas/minúsculas

- Ejemplo: AD tiene `administrador`, catálogo tiene `Administrador`
- **Acción**: Se omite (no coincide case-sensitive); usuario no obtiene el permiso
- **Solución**: Cliente debe renombrar a `Administrador` en su AD

Escenario 3: El cliente requiere un nuevo rol

- Si el cliente necesita un rol adicional (ej: `GestorFinanciero`)
- **Proceso**:
  a. Cliente contacta a la plataforma solicitando el nuevo rol
  b. Plataforma evalúa y crea el rol en el sistema
  c. Plataforma actualiza el catálogo de roles
  d. Cliente descarga el catálogo actualizado
  e. Cliente crea el grupo en su AD con el nombre exacto
  f. Se sincroniza

---

7. Coexistencia de Usuarios Locales y AD

7.1 Arquitectura Multitenant

Modelo de Datos:

- Un usuario puede estar asociado a múltiples clientes (tenants)
- Cada asociación usuario-cliente tiene su propia configuración de autenticación:
  - Cliente A → Usuario autentica con AD
  - Cliente B → Usuario autentica localmente (usuario/contraseña)

7.2 Determinación del Método de Autenticación

En el Login:

1. Usuario selecciona cliente (tenant)
2. Sistema consulta la configuración
3. Si es "AD": redirige a flujo SAML
4. Si es "Local": muestra formulario de usuario/contraseña

---

8. Segmentación Local (Prefijos y Niveles)

8.1 Alcance en Integración SCIM

Gestión Exclusivamente Local:

- La segmentación (prefijos y niveles 1-4) se gestiona **exclusivamente** en la plataforma
- No se sincroniza desde el AD en el alcance inicial
- Los prefijos se asignan manualmente a usuarios en la interfaz de administración de la plataforma

Separación de Responsabilidades:

- **AD gestiona**: Usuarios, grupos, roles (quién puede acceder a qué módulos)
- **Plataforma gestiona**: Segmentación por prefijos (qué información ve dentro de los módulos)

8.2 Diseño Preparado para Futura Extensión

Aunque no se implementa en el alcance inicial, el diseño debe permitir futura sincronización de segmentación desde el AD:

Consideraciones de Diseño:

- La tabla de segmentación debe incluir un campo `Origen` (Local / AD)
- Si en el futuro se implementa sincronización de segmentación desde AD:
  - Prefijos con origen "AD" no pueden ser editados localmente
  - Prefijos con origen "Local" se mantienen editables
- El endpoint SCIM debe estar preparado para recibir atributos personalizados (ej: `prefijos`, `nivel_segmentacion`)

---

9. Manejo de Errores y Resiliencia

9.1 Principios de Diseño

El sistema debe ser:

- **Idempotente**: Operaciones pueden ejecutarse múltiples veces con el mismo resultado
- **Resiliente**: Capaz de manejar errores transitorios sin fallos críticos
- **Observable**: Registra eventos y errores para monitoreo y troubleshooting

9.2 Reintentos Automáticos

Errores Transitorios:

- Timeouts de red
- Errores de disponibilidad (base de datos temporalmente no disponible)
- Sobrecarga del sistema

9.3 Registro de Eventos Fallidos

Persistencia de Fallos:

- Todas las operaciones SCIM fallidas se registran en una tabla de auditoría
- Información registrada:
  - Timestamp
  - Tipo de operación (POST, PATCH, DELETE)
  - Recurso afectado (ID de usuario, email)

- Payload completo del request SCIM
- Tipo de error
- Mensaje de error
- Código HTTP de respuesta
- Cliente (tenant) afectado

9.4 Alertas y Monitoreo

Alertas Automáticas:

- Sistema envía alertas al equipo operativo ante:
  - Múltiples fallos consecutivos de sincronización (ej: 5 fallos en 10 minutos)
  - Fallo en autenticación del SCIM client (Bearer token inválido)
  - Tasa de error superior al 5% en eventos SCIM
  - Falta de sincronización por más de 1 hora (si se esperaban eventos)

Canales de Alerta:

- Email a administradores

---

10. Logs y Auditoría de Sincronización

10.1 Registro de Eventos SCIM

Obligatorio: Registrar Cada Evento SCIM Recibido

El sistema debe registrar TODOS los eventos SCIM recibidos, exitosos o fallidos.

Información Registrada:

- Timestamp de recepción
- Cliente (tenant) que envía el evento
- Tipo de operación (POST / PATCH / DELETE)
- ID del usuario afectado
- Email del usuario afectado
- Payload completo del request SCIM (JSON)
- Resultado de la operación (Éxito / Error)
- Código HTTP de respuesta
- Mensaje de error (si aplica)
- Duración del procesamiento (en milisegundos)
- Roles procesados (validados contra catálogo)
- Roles omitidos (no encontrados en catálogo)

Formato de Log (Ejemplo JSON):

10.2 Historial de Cambios de Usuario

Obligatorio: Mantener Historial de Cambios (Quién, Qué, Cuándo)

El sistema debe mantener un historial completo de cambios para cada usuario:

10.3 Auditoría de Invalidación de Sesiones

Registro de Sesiones Invalidadas:

Cada vez que se invalida una sesión proactivamente, se almacenan los datos correspondientes:

- Identificador del usuario
- Identificador del cliente al que estaba asciada la sesión
- timestamp de la invalidación de la sesión

- Razón de la invalidación (deshabilitacion_usuario / eliminacion_usuario / cambio_roles / cambio_grupos)
- Referencia al evento que generó la invalidación

Uso:

- Permite auditar cuántas sesiones se cerraron por cambios de permisos
- Útil para investigaciones de seguridad
- Permite medir efectividad de la invalidación proactiva

10.4 Retención de Logs

Políticas de Retención:

- Logs de eventos SCIM: **2 años**
- Historial de cambios de usuario: **5 años** (cumplimiento normativo)
- Auditoría de sesiones invalidadas: **2 años**
- Logs de errores: **1 año**

Almacenamiento:

- Base de datos principal para logs recientes (últimos 6 meses)
- Almacenamiento en frío (S3, Azure Blob) para logs antiguos
- Compresión de logs antiguos para optimizar espacio

---

11. Configuración Multicliente

11.1 Endpoint SCIM Único con Diferenciación por Tenant

Nota: en las rutas, plataforma será reemplazado por el nombre de la plataforma

Ruta del Endpoint:

```
https://plataforma.com/scim/v2/{tenantId}/Users
```

Parámetros:

- `{tenantId}`: Identificador único del cliente (UUID o código alfanumérico)

Ejemplos:

```
POST https://plataforma.com/scim/v2/cliente-abc-123/Users
PATCH https://plataforma.com/scim/v2/cliente-abc-123/Users/usuario-456
DELETE https://plataforma.com/scim/v2/cliente-abc-123/Users/usuario-456
```

Validación:

- Sistema valida que el `tenantId` exista en la base de datos
- Sistema valida que el Bearer token corresponda a ese `tenantId`
- Si no coincide, responde con HTTP 403 Forbidden

11.2 Configuración por Cliente

Cada cliente tiene su propia configuración almacenada en la base de datos:

Ejemplo Tabla de Configuración:

```

text
Tabla: ClienteConfiguracion
- ID
- ClienteID (tenantId)
- NombreCliente
- IntegracionADActiva (True / False)
- URLIdentityProvider (URL del IdP SAML)
- CertificadoSAML (certificado público del IdP)
- BearerTokenSCIM (token encriptado)
- FechaExpiracionToken
- GruposRolesFiltro (JSON: lista de grupos/roles a sincronizar)
- DuracionSesion (en horas, default: 4)
- SegmentacionActiva (True / False)
```

```
- NivelesSegmentacionUsados (JSON: [1, 2, 3, 4])
- FechaCreacion
- FechaUltimaModificacion

```

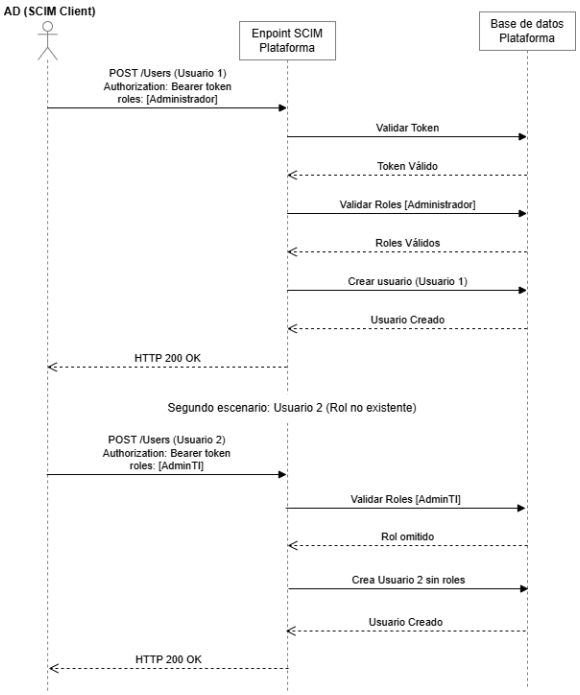
Ejemplo:

```

json
{
  "clienteID": "cliente-abc-123",
  "nombreCliente": "Empresa ABC",
  "integracionADActiva": true,
  "urlIdentityProvider": "https://adfs.empresa-abc.com/adfs/ls",
  "certificadoSAML": "-----BEGIN CERTIFICATE-----...",
  "bearerTokenSCIM": "encrypted_token_here",
  "fechaExpiracionToken": "2026-04-15",
  "gruposRolesFiltro": ["Administrador", "Gestor", "Usuario"],
  "duracionSesion": 4,
  "segmentacionActiva": true,
  "nivelesSegmentacionUsados": [1, 2, 3]
}

```

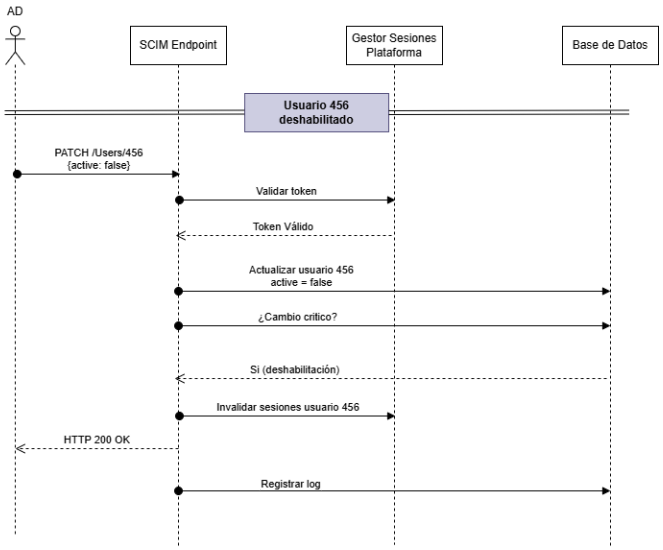
---

12. Flujo Completo de Integración (Diagrama de Secuencia)

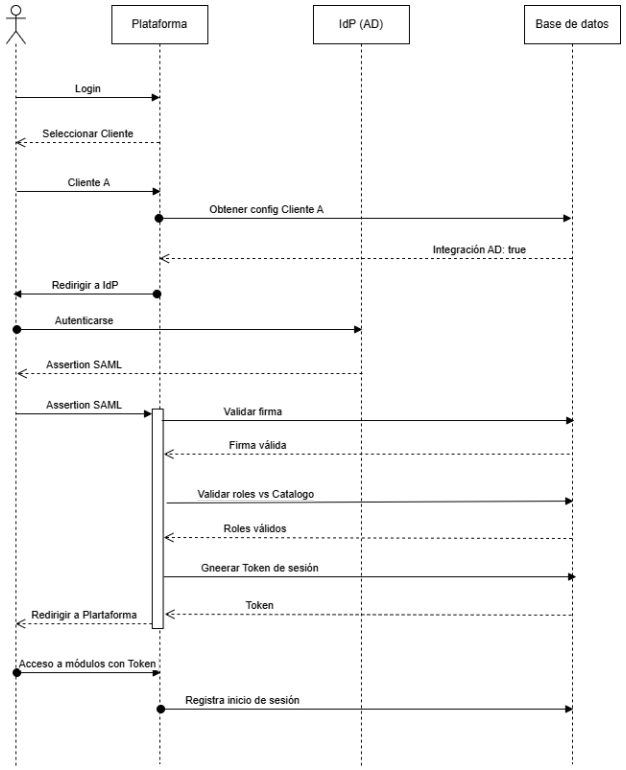
Sincronización Inicial



Evento de Cambio en Tiempo Real (Deshabilitación)



Autenticación SAML con Usuario Sincronizado



---

13. Criterios de Aceptación

13.1 Sincronización SCIM

- El endpoint SCIM responde correctamente a operaciones POST, PATCH, DELETE
- Eventos SCIM se procesan en menos de 500ms (percentil 95)
- Operaciones SCIM son idempotentes
- Bearer token se valida correctamente en cada request
- Roles/grupos recibidos se validan contra el catálogo de roles de la plataforma
- Validación es exacta y case-sensitive: `Administrador` ≠ `administrador`
- Roles que no existen en catálogo se omiten silenciosamente
- Usuarios sin roles válidos se crean sin permisos
- Sistema soporta múltiples clientes con diferentes ADs

- Sincronización inicial importa solo usuarios de grupos/roles especificados
- Roles omitidos se registran en logs con detalle

13.2 Autenticación SAML

- Flujo SAML 2.0 funciona correctamente con IdP de prueba
- Assertion SAML se valida correctamente (firma, expiración)
- Roles se extraen de la assertion y se validan contra catálogo
- Validación de roles es exacta y case-sensitive
- Roles no encontrados en catálogo se omiten silenciosamente
- Token de sesión tiene duración de 4 horas
- Usuario puede seleccionar cliente y el sistema determina método de autenticación
- Usuarios locales y usuarios AD coexisten sin conflictos
- Usuario local puede migrar a AD sin perder datos

13.3 Invalidación de Sesiones

- Deshabilitación de usuario invalida sesión en menos de 1 minuto
- Eliminación de usuario invalida sesión en menos de 1 minuto
- Remoción de rol invalida sesión en menos de 1 minuto
- Remoción de grupo invalida sesión en menos de 1 minuto
- Usuario recibe mensaje claro al intentar usar sesión invalidada
- Usuario debe reautenticarse con permisos actualizados

13.4 Manejo de Errores

- Errores transitorios devuelven HTTP 5xx
- SCIM client ejecuta reintentos automáticos ante HTTP 5xx
- Eventos fallidos se registran en tabla de auditoría
- Administradores pueden ver lista de eventos fallidos
- Sistema envía alertas ante múltiples fallos consecutivos

13.5 Logs y Auditoría

- Todos los eventos SCIM se registran (exitosos y fallidos)
- Historial de cambios registra quién, qué y cuándo
- Sesiones invalidadas se registran con razón de invalidación
- Logs incluyen roles omitidos (no encontrados en catálogo)
- Logs se retienen según políticas definidas
- Administradores pueden buscar y filtrar logs

13.6 Gestión de Catálogo de Roles

- Catálogo de roles está disponible para descargar en administración
- Catálogo exportable en formato CSV/JSON
- Interfaz proporciona guía clara para cliente sobre coincidencia de nombres
- Documentación menciona la responsabilidad del cliente sobre nombres exactos

13.7 Segmentación

- Segmentación se gestiona localmente, independiente de sincronización SCIM
- Usuarios sincronizados vía SCIM pueden tener segmentación asignada
- Segmentación se aplica después de validar roles
- Sistema está preparado para futura sincronización de segmentación desde AD

---

14. Consideraciones de Seguridad

14.1 Protección del Endpoint SCIM

- **HTTPS obligatorio**: El endpoint SCIM solo acepta conexiones HTTPS (TLS 1.2 o superior)
- **Autenticación obligatoria**: Todos los requests requieren Bearer token válido
- **Rate limiting**: Máximo 100 requests por minuto por cliente (configurable)
- **Validación de payloads**: Validar estructura JSON según esquema SCIM 2.0
- **Sanitización de inputs**: Prevenir inyección SQL, XSS, etc.

14.2 Gestión de Tokens

- **Encriptación**: Bearer tokens se almacenan encriptados en base de datos
- **Transmisión segura**: Tokens solo se envían vía HTTPS
- **Rotación periódica**: Tokens se rotan cada 90 días
- **Expiración**: Tokens tienen fecha de expiración
- **Auditoría**: Uso de tokens se registra en logs

14.3 Validación de Assertions SAML

- **Validación de firma**: Verificar firma digital de la assertion
- **Validación de certificado**: Verificar que certificado del IdP sea válido y no haya expirado
- **Validación de expiración**: Verificar que assertion no haya expirado (NotBefore, NotOnOrAfter)
- **Validación de audiencia**: Verificar que la assertion esté dirigida a esta plataforma
- **Prevención de replay attacks**: Validar que assertion no haya sido usada antes

14.4 Protección de Sesiones

- **Tokens firmados**: Tokens JWT se firman criptográficamente
- **Duración limitada**: Sesiones expiran después de 4 horas
- **Invalidación proactiva**: Sesiones se invalidan ante cambios críticos
- **Lista negra**: Tokens invalidados se agregan a blacklist
- **Transmisión segura**: Tokens solo se transmiten vía HTTPS

---

15. Glosario

- **AD**: Active Directory
- **SCIM**: System for Cross-domain Identity Management
- **SAML**: Security Assertion Markup Language
- **SSO**: Single Sign-On
- **IdP**: Identity Provider
- **JWT**: JSON Web Token
- **Bearer Token**: Token de autenticación portador
- **Assertion**: Declaración de identidad firmada por el IdP
- **Tenant**: Cliente en arquitectura multicliente
- **Soft Delete**: Eliminación lógica (marcar como eliminado sin borrar)
- **Idempotencia**: Propiedad de operaciones que pueden ejecutarse múltiples veces con el mismo resultado
- **Backoff Exponencial**: Estrategia de reintentos con esperas cada vez mayores
- **Catálogo de Roles**: Lista de roles disponibles en la plataforma que el cliente debe replicar en su AD
- **Replicación Directa**: Estrategia donde el cliente crea en AD grupos/roles con exactamente los mismos nombres de la plataforma

---

Anexo A: Ejemplo de Payload SCIM

Crear Usuario (POST)

```
{
  "schemas": ["urn:ietf:params:scim:schemas:core:2.0:User"],
  "userName": "usuario@empresa.com",
  "name": {
    "givenName": "Juan",
    "familyName": "Pérez"
  },
  "emails": [
    {
      "value": "usuario@empresa.com",
      "type": "work",
      "primary": true
    }
  ],
  "active": true,
  "groups": [
    {
      "value": "Administrador",
      "display": "Administrador"
    },
    {
      "value": "Auditor",
      "display": "Auditor"
    }
  ]
}

```

Actualizar Usuario (PATCH)

```
{
  "schemas": ["urn:ietf:params:scim:api:messages:2.0:PatchOp"],
  "Operations": [
    {
      "op": "replace",
      "path": "active",
      "value": false
    }
  ]
}
```

Eliminar Usuario (DELETE)

```
DELETE /scim/v2/cliente-abc-123/Users/usuario-456
Authorization: Bearer <token>
```

Anexo B: Guía de Configuración para el Cliente

Paso 1: Descargar Catálogo de Roles

1. Administrador de la plataforma envía link para descargar catálogo
2. Catálogo contiene lista de roles con nombres exactos
3. Ejemplo:

```
Administrador
Auditor
Analista
Gestor
Supervisor
Usuario
```

Paso 2: Verificar/Crear Grupos en AD

1. Cliente abre su Active Directory
2. Revisa si existen grupos con exactamente los mismos nombres del catálogo
3. Si faltan grupos, los crea
4. **Importante**: Los nombres deben coincidir exactamente (case-sensitive)
   - Correcto: `Administrador`
   - Incorrecto: `administrador` (minúscula)
   - Incorrecto: `Admin_TI` (nombre diferente)

Paso 3: Asignar Usuarios a Grupos

1. Cliente asigna usuarios del AD a los grupos creados
2. Usuarios pueden estar en múltiples grupos

Paso 4: Configurar SCIM Client en AD

1. Cliente proporciona endpoint SCIM y token a su administrador de AD
2. Administrador configura SCIM client en su AD
3. Endpoint: `https://plataforma.com/scim/v2/{tenantId}/Users`
4. Token: [Bearer token generado por plataforma]

Paso 5: Validación Previa

1. Cliente realiza prueba de sincronización en ambiente de staging
2. Verifica que los usuarios se sincronicen correctamente
3. Verifica que los roles sean validados correctamente
4. Ajusta nombres en AD si es necesario

Paso 6: Sincronización Inicial

1. Cliente inicia sincronización desde su AD
2. Plataforma valida cada rol contra catálogo
3. Usuarios se crean con roles válidos
4. Usuarios con roles inválidos se crean sin permisos (revisión manual requerida)