

Documentación

# Solución de problemas

Solución de problemas

Anonymous · 11/09/2026

---

# Tabla de contenidos

|                                                              |    |
|--------------------------------------------------------------|----|
| <b>Solución de problemas</b>                                 | 3  |
| <b><i>Errores 401 al consumir la API</i></b>                 | 4  |
| <b><i>Errores 403: sin acceso al recurso</i></b>             | 6  |
| <b><i>Errores 402 y conexión suspendida</i></b>              | 8  |
| <b><i>Errores de proxy y timeouts hacia el SII</i></b>       | 10 |
| <b><i>Errores 400: estructura del request</i></b>            | 13 |
| <b><i>Recibo HTML o "Error interno" en vez de JSON</i></b>   | 17 |
| <b><i>Errores 409, lentitud e intermitencias del SII</i></b> | 19 |
| <b><i>Errores 429 y bloqueos por exceso de uso</i></b>       | 23 |

Documentación » Solución de problemas

# Solución de problemas

Solución de problemas

Anonymous · 11/09/2026

---

## Solución de problemas

Esta sección está organizada **por síntoma**: lo que ves cuando algo falla, y qué revisar en cada caso.

Si prefieres partir del código de respuesta, la [tabla de códigos](#) los enumera todos con su significado.

### Los cinco culpables habituales

Antes de un diagnóstico largo, descarta estos cinco en orden. Resuelven la mayoría de los casos:

1. **¿Es el día 1 del mes o pasó hace poco?** Puede ser una suspensión por créditos.
2. **¿Estás en la plataforma correcta?** V2 y Legacy son servicios independientes.
3. **¿Están activos los productos?** Y presionaste “Actualizar productos y límites”.
4. **¿Está el proxy configurado y respondiendo?**
5. **¿La estructura del request coincide campo por campo con la documentación?**

---

Última actualización el 11/09/2026

Documentación » Solución de problemas » Errores 401 al consumir la API

## Errores 401 al consumir la API

Diagnóstico de los distintos 401: token inválido, cabecera mal formada, credenciales del SII y sesión caída.

Anonymous · 11/09/2026

## Errores 401 al consumir la API

El 401 agrupa cuatro problemas distintos. El mensaje del campo `detail` te dice cuál es.

### “Las credenciales de autenticación no se proveyeron”

Falta la palabra `Token` en la cabecera. La API **no acepta** `Bearer`, ni `X-Api-Key`, ni el token en la URL:

|                                                                             |              |
|-----------------------------------------------------------------------------|--------------|
| <code>Authorization: Token 0123456789abcdef0123456789abcdef01234567</code>  | ← correcto   |
| <code>Authorization: Bearer 0123456789abcdef0123456789abcdef01234567</code> | ← incorrecto |
| <code>Authorization: 0123456789abcdef0123456789abcdef01234567</code>        | ← incorrecto |

### “Token de conexión inválido”

El token no corresponde a ninguna conexión activa. Revisa en orden:

1. **Compáralo carácter por carácter** con el de [Gestionar conexión → Token](#). En la mayoría de los casos el token de la integración quedó desactualizado o se cargó en el lugar equivocado del código.
2. **¿Lo regeneraste?** El anterior deja de funcionar de inmediato.
3. **¿Es el token de la conexión** y no la API Key de tu usuario?
4. **¿Estás usando la plataforma correcta?** El token de V2 no funciona en Legacy.
5. **¿La URL base es la correcta?** `https://app.apigateway.cl/api/v2/`, no `apigateway.cl`.

### “La conexión está inactiva”

El token está bien: la conexión está **suspendida**. Revisa [Créditos y conexión suspendida](#).

## Credenciales del SII incorrectas

Cuando el 401 incluye un código de error del propio SII:

```
{
  "detail": "En la respuesta del SII se obtuvo el código de error #612. Descripción del error oficial del SII: Contraseña incorrecta."
}
```

el problema está en las credenciales que envías en el cuerpo, no en la plataforma.

1. **Verifica la clave** ingresando directamente a [www.sii.cl](http://www.sii.cl).
2. **Verifica el formato del RUT**: sin puntos, con guión, `11111111-1`. Un RUT con puntos puede producir errores que parecen de otra naturaleza, incluidos timeouts de navegación.
3. **Verifica la estructura de auth**: debe ser exactamente `auth.pass.rut` y `auth.pass.clave`.
4. **Si cambiaste la clave recientemente**, actualízala en tu integración y fuerza una sesión nueva con `auth_cache=0` en la primera consulta.

## “Se debe volver a autenticar al usuario en la sesión del SII”

---

La sesión reutilizada quedó en mal estado. Puede venir acompañado de la cabecera `X-Auth-Session-Problem: 1`.

Agrega `auth_cache=0` a la URL en esa consulta, y vuelve a operar sin el parámetro después.

### No lo dejes puesto en producción

Cada `auth_cache=0` equivale a un login nuevo en el SII. Usarlo en todas las peticiones aumenta el riesgo de que el SII bloquee tus accesos.

## 401 en un recurso específico, mientras otros funcionan

---

Si un recurso devuelve `401` de forma reproducible y otros funcionan con las mismas credenciales, revisa el **método de autenticación que ese recurso requiere**. Algunos exigen certificado digital y no RUT y clave.

Un caso real: `bienes_raices/propiedades/contribuyente` devolvía `401` en contribuyentes con muchos roles inmobiliarios al consultarlo con RUT y clave. Ese recurso requiere **certificado digital**.

Revisa [Token y certificado](#) y la documentación del recurso en </docs/api>.

---

Última actualización el 11/09/2026

Documentación » Solución de problemas » Errores 403: sin acceso al recurso

## Errores 403: sin acceso al recurso

Diagnóstico cuando la conexión no tiene productos asociados o no tiene acceso a una operación.

Anonymous · 11/09/2026

## Errores 403: sin acceso al recurso

Hay dos mensajes y significan cosas distintas.

### “La conexión no tiene productos asociados”

```
{ "status": "403", "detail": "La conexión no tiene productos asociados." }
```

Tu conexión **no tiene ningún producto activo**. El token es válido y la petición está bien formada, pero no hay acceso a ningún recurso.

Actívalos en [Gestionar conexión → Productos](#).

### “La conexión no tiene acceso a la operación X”

```
{ "status": "403", "detail": "La conexión no tiene acceso a la operación  
sii.contribuyentes.situacion_tributaria.get_tercero." }
```

Sí tienes productos activos, pero **ninguno incluye esa operación**. El prefijo del código te dice qué producto falta:

| Prefijo              | Producto                   |
|----------------------|----------------------------|
| sii.contribuyentes.* | Info. Contribuyentes       |
| sii.rcv.*            | Registro de Compra y Venta |
| sii.dte.*            | Documentos Tributarios     |
| sii.rtc.*            | Cesión de DTE              |
| sii.bhe.*            | Boletas de Honorarios      |
| sii.bte.*            | Boletas de Terceros        |
| sii.mipyme.*         | Portal MiPYME              |
| sii.eboleta.*        | Portal eBoleta             |
| sii.f29.*            | Formulario 29              |

| Prefijo                | Producto                  |
|------------------------|---------------------------|
| sii.bienes_raices.*    | Bienes Raíces             |
| sii.vehiculos.*        | Vehículos                 |
| previred.indicadores.* | Indicadores Previsionales |

Tener activo un producto **no habilita** los recursos de otro: cada producto habilita solo los suyos.

## “Activé el producto y sigue dando 403”

Es el reporte más frecuente de esta categoría, y casi siempre la selección **no se guardó**.

En la pestaña de productos, marcar los casilleros no activa nada: hay que presionar **“Actualizar productos y límites”** al final del formulario.

**Cómo comprobarlo:** revisa el [historial de eventos de la conexión](#). Cada activación queda registrada con fecha, hora y créditos descontados. Si no hay registro, la activación nunca ocurrió.

Otras causas posibles:

- **No alcanzaron los créditos.** Activar un producto cobra de inmediato; si el saldo no cubre el costo, la actualización se rechaza.
- **Activaste el producto equivocado.** El mensaje nombra la operación exacta: contrástala con la tabla de arriba.

### No hay que regenerar el token tras activar un producto

El token está asociado a la conexión, no a los productos, y la pestaña *Token* no muestra qué servicios incluye. Regenerarlo no resuelve un 403 y obliga a actualizar tu integración sin necesidad.

Última actualización el 11/09/2026

Documentación » Solución de problemas » Errores 402 y conexión suspendida

## Errores 402 y conexión suspendida

Diagnóstico cuando faltan créditos, la conexión está suspendida o la IP está en uso por otra conexión.

Anonymous · 11/09/2026

## Errores 402 y conexión suspendida

### “La conexión está inactiva”

```
{ "detail": "La conexión está inactiva." }
```

La conexión está **suspendida por créditos insuficientes**. Ocurre cuando el **día 1 del mes** el saldo no alcanza para cubrir la **base mensual completa** de los productos activos más el pack de usuarios.

Casos reales de este escenario: saldo de 20,018 con base de 25; saldo de 12,316 con base de 15,600; saldo de 3.425 con una base mayor. En todos hay créditos, pero no los suficientes.

**Solución:** compra créditos en [Gestionar conexión → Créditos](#). Al procesarse el pago, el sistema descuenta el **proporcional de los días que quedan** del mes y reactiva la conexión **automáticamente**. No requiere intervención de soporte.

**Cuánto necesitas:** tu base mensual está en [Gestionar conexión → Productos](#), en “Base mensual estimada (tu selección)”.

### “Créditos insuficientes para realizar la consulta”

```
{
  "status": 402,
  "detail": "Créditos insuficientes para realizar la consulta. Créditos disponibles:
0.101, costo de esta operación: 0.6"
}
```

El mensaje incluye ambos valores, así que sabes exactamente cuánto falta.

Si el costo indicado es **mayor** al que esperabas según el cotizador, lo más probable es un **usuario de autenticación adicional**: 0,6 créditos por cada credencial nueva del SII, una vez por período. Revisa [Productos y límites](#).

## “No está permitido realizar consultas desde la IP...”

---

```
{
  "status": "402",
  "detail": "No está permitido realizar consultas desde la IP 203.0.113.99, ya que se encuentra en uso por otra conexión. Para consultar desde esta IP debe realizar la compra de créditos."
}
```

Tu conexión **nunca ha pagado un cobro**: opera con créditos de bienvenida. Esas conexiones solo pueden consultar desde una IP que **ninguna otra conexión** haya usado.

**Solución:** compra créditos. Con el pago registrado, la restricción se levanta y la conexión puede consultar desde cualquier IP, incluidas las compartidas. No se libera por soporte.

**Mientras tanto:** consulta desde una IP que no esté en uso por otra conexión.

Este mensaje también aparece al usar una misma conexión desde dos ambientes con IP de salida distinta antes de haber pagado.

### No crees varias cuentas para obtener más créditos de prueba

Crear cuentas o conexiones adicionales para acumular créditos de bienvenida provoca el **bloqueo de las cuentas involucradas**, conforme a los [términos y condiciones](#).

## Un descuento de créditos que no cuadra

---

Revisa en este orden:

1. **Usuario de autenticación adicional** (+0,6). La causa más frecuente.
2. **El cargo fijo mensual** del día 1.
3. **La activación de un producto**, proporcional a los días restantes del mes.

Las tres quedan registradas con su monto exacto en el [historial de eventos de la conexión](#).

---

Última actualización el 11/09/2026

Documentación » Solución de problemas » Errores de proxy y timeouts hacia el SII

## Errores de proxy y timeouts hacia el SII

Qué revisar cuando la API no logra salir al SII por el proxy de la conexión, incluidas las consultas que quedan colgadas y terminan canceladas por tu cliente.

Anonymous · 11/09/2026

## Errores de proxy y timeouts hacia el SII

El proxy es un **requisito obligatorio** del servicio. Cuando está ausente o mal configurado, la API no logra salir al SII y la consulta falla.

### Mensajes que indican un problema de proxy

Estos mensajes vienen del propio control de proxy de la API y apuntan directamente a él:

```
{ "detail": "Error de conexión con el proxy. Verifique la configuración del proxy.
Detalle: ... ProxyError('Unable to connect to proxy', ... [Errno 111] Connection
refused)" }
```

```
{ "detail": "Error de conexión con el proxy. Verifique la configuración del proxy.
Detalle: ... [Errno 101] Network is unreachable" }
```

Estos otros indican que la salida hacia el SII no se completó, y la causa habitual es el proxy:

```
{ "message": "cURL error 28: Connection timed out after 20001 milliseconds" }
```

```
{ "message": "cURL error 97: Can't complete SOCKS5 connection to zeusr.sii.cl" }
```

También puede manifestarse de una forma menos evidente: **consultas que quedan procesando varios minutos y terminan en 400**, con tiempos muy parecidos entre sí, porque la comunicación con el SII nunca se completa.

### Cuando no hay ningún mensaje de error

Es el caso más difícil de diagnosticar, y el que más se confunde con lentitud del SII: la petición sale de tu aplicación, **la API nunca responde**, y al cabo de varios minutos es tu propio cliente el que cancela.

```
The request was canceled due to the configured HttpClient.Timeout of 180 seconds
```

```
elapsing.
```

Ese error lo genera tu cliente, no la API, así que no trae ninguna pista de la causa. La primera sospecha es el proxy.

Lo determinante es **cómo** falla:

| Cómo falla el proxy                                                                  | Qué ves                                                        |
|--------------------------------------------------------------------------------------|----------------------------------------------------------------|
| Rechaza la conexión: el servicio está caído o el puerto cerrado                      | Un error rápido y explícito: <code>Connection refused</code>   |
| Descarta los paquetes: un firewall o un grupo de seguridad que bloquea sin responder | <b>Nada.</b> La petición queda colgada hasta que alguien corta |

Un bloqueo silencioso no produce error: produce espera. Por eso un proxy inalcanzable se ve igual que un SII lento, y por eso **subir el timeout no sirve** en este caso.

### Estos otros no son exclusivamente de proxy

Los errores de navegación o de autenticación en los portales del SII —por ejemplo `Page.goto: Timeout ... exceeded` o `Timeout esperando localStorage` en el flujo OAuth de eBoleta— pueden originarse en el proxy, pero también en el formato del RUT enviado, en los permisos del usuario ante el SII o en una intermitencia del propio SII. Descarta el proxy con los pasos de abajo y, si persiste, revisa [Errores 400: estructura del request](#) y la documentación del recurso.

## Cómo verificar si el proxy está operando

Revisa la cabecera `X-Stats-HttpClientProxy` en la respuesta:

| Valor | Significado                       |
|-------|-----------------------------------|
| 1     | La consulta salió por tu proxy    |
| 0     | El proxy <b>no</b> se está usando |

Si vale 0, revisa que la URL esté bien guardada en tu conexión y que tenga el formato correcto:

```
http://<usuario>:<contraseña>@<ip_del_proxy>:3128
```

## Lista de verificación

1. **El contenedor está arriba:** `docker compose ps` y `docker compose logs squid`.
2. **El puerto 3128 es accesible desde Internet:** revisa firewall del sistema y del proveedor (grupos de seguridad, reglas de red).

3. **Las credenciales coinciden:** el usuario y la contraseña de la URL configurada en API Gateway deben ser exactamente los del archivo de contraseñas del proxy.
4. **La IP no cambió:** si tu servidor cambió de IP, la URL del proxy quedó apuntando a la anterior.
5. **Los dominios permitidos están completos:** `.sii.cl`, `.amazonaws.com` y `.previred.com`. Deben estar los tres: comprobar solo los del SII no basta.
6. **Nuestra IP está permitida**, si restringes por origen: obténla con `nslookup app.apigateway.cl`.

Prueba tu proxy por tu cuenta desde cualquier máquina, **fuera de la red donde está instalado**. Desde dentro puede responder aunque no sea alcanzable desde Internet.

Primero comprueba que el puerto responda, sin meter HTTP de por medio:

```
nc -vz -w 5 TU_IP 3128
```

| Resultado           | Qué significa                                                                           |
|---------------------|-----------------------------------------------------------------------------------------|
| succeeded!          | El puerto es alcanzable. Sigue con la prueba de tráfico                                 |
| Operation timed out | Nadie contesta: algo descarta los paquetes. Revisa firewall del sistema y del proveedor |
| Connection refused  | El puerto se alcanza, pero no hay nada escuchando. El proxy está caído                  |

Después prueba el tráfico:

```
curl -x http://usuario:contrasena@TU_IP:3128 https://www.sii.cl -I --max-time 15
```

Deberías ver `HTTP/1.1 200 Connection established`. Si en cambio obtienes `curl: (28) Connection timed out`, tu proxy no es alcanzable desde fuera: es el mismo diagnóstico del `Operation timed out` de arriba.

Si cualquiera de las dos pruebas falla, el problema está en tu proxy y no en API Gateway.

## Un patrón que confunde

Es habitual que el proxy responda bien a tus pruebas y que, sin embargo, **sus logs no muestren ninguna conexión entrante** desde nuestros servidores hacia dominios del SII. Cuando ocurre eso, la conexión no está llegando a tu proxy: revisa firewall, grupos de seguridad y reglas de red del proveedor, más que la configuración de Squid.

El [tutorial de proxy](#) trae el `docker-compose.yml` y el `squid.conf` listos, y sus [problemas comunes](#) el detalle de cada punto.

Documentación » Solución de problemas » Errores 400: estructura del request

## Errores 400: estructura del request

Los siete errores de estructura que explican la mitad de los problemas reportados, con ejemplos correctos e incorrectos.

Anonymous · 11/09/2026

## Errores 400: estructura del request

La mitad de los problemas que atendemos son errores 400 causados por la estructura del cuerpo de la petición. Esta es la lista de los patrones concretos que hemos visto, y sirve como verificación antes de reportar un caso.

### 1. Campos en el nivel equivocado

Los datos deben ir anidados donde el recurso los espera, no en la raíz del JSON.

**Incorrecto** — datos del DTE al nivel raíz:

```
{
  "emisor": "22222222-2",
  "receptor": "33333333-3",
  "dte": 33,
  "folio": 93,
  "auth": { "pass": { "rut": "11111111-1", "clave": "..." } }
}
```

**Correcto** — anidados dentro de `dte`:

```
{
  "auth": { "pass": { "rut": "11111111-1", "clave": "..." } },
  "dte": {
    "dte": 33,
    "emisor": "22222222-2",
    "receptor": "33333333-3",
    "folio": 93,
    "fecha": "2026-04-13",
    "total": 1407734
  }
}
```

Lo mismo ocurre con los **filtros** de los recursos del Portal MiPYME, que van dentro de un objeto `filtros`:

**Incorrecto:**

```
{
  "auth": { "pass": { "rut": "11111111-1", "clave": "..." } },
  "FEC_DESDE": "2026-08-01",
  "FEC_HASTA": "2026-08-25",
  "TPO_DOC": "33"
}
```

**Correcto:**

```
{
  "auth": { "pass": { "rut": "11111111-1", "clave": "..." } },
  "filtros": {
    "FEC_DESDE": "2026-08-01",
    "FEC_HASTA": "2026-08-31",
    "TPO_DOC": 33,
    "ESTADO": "RRC"
  }
}
```

Este error en particular se manifiesta como un 409 de indisponibilidad del SII, no como un 400, lo que lo hace difícil de diagnosticar. Revisa [Intermitencias del SII](#).

## 2. Formato de fechas

Siempre **ISO**, **YYYY-MM-DD**:

| Correcto   | Incorrecto |
|------------|------------|
| 2026-06-21 | 21-06-2026 |

## 3. Formato de períodos

Los períodos mensuales van como **YYYYMM**, **sin guión**:

| Correcto | Incorrecto |
|----------|------------|
| 202605   | 2026-05    |

Enviarlo con guión produce un 404 **Página no encontrada**, porque la ruta no calza. Algunos listados aceptan día en formato **YYYYMMDD**: confirma en la documentación del recurso.

## 4. Formato del RUT

Sin puntos, con guión y con dígito verificador, y la K en mayúscula:

| Correcto   | Incorrecto   |
|------------|--------------|
| 11111111-1 | 76.192.083-9 |
| 22222222-2 | 22222222-2   |

Un RUT con puntos puede producir errores que parecen de otra naturaleza. Un caso real recibía un timeout de navegación al portal del SII cuya causa era "rut": "76.491.518-6".

## 5. Confundir el RUT de cada campo

Es frecuente confundir el RUT del **contribuyente que se consulta** con el de la **credencial que inicia sesión**. En muchos recursos son distintos:

- En la **ruta**: el RUT del contribuyente cuya información pides.
- En `auth.pass.rut`: el RUT de quien **inicia sesión**, que puede ser una persona natural autorizada.

En el recurso de emisión de eBoleta hay tres RUT y cada uno cumple un rol distinto:

| Campo                                          | Qué debe contener                       |
|------------------------------------------------|-----------------------------------------|
| <code>auth.pass.rut</code>                     | El usuario autorizado que inicia sesión |
| <code>dte.vendedor</code>                      | El <b>mismo</b> usuario que autentica   |
| <code>dte.Encabezado.Emissor.RUTEmissor</code> | La <b>empresa</b> emisora               |

Enviar el RUT del usuario donde va la empresa produce el error "No tiene folios asignados", que parece un problema tributario y no lo es.

## 6. Parámetros que no existen

Agregar campos que el recurso no define —por ejemplo un `receptor_rut` inventado— puede hacer fallar la operación. Envía solo lo documentado.

## 7. Parámetros obligatorios ausentes o con valores inválidos

Algunos listados requieren parámetros fáciles de pasar por alto, como `pagina`. El mensaje los nombra: "El parámetro 'pagina' es requerido".

Y un valor fuera de la lista válida produce error aunque el JSON esté bien formado. Por ejemplo, en

documentos recibidos del Portal MiPYME los estados válidos son EMI, PRV, DCD, DEI, DFI, DIN, DPF, DRF, DRI, DRR, INI, RAC, RAD, RNR, RRC, RRH y RSR. Un valor como RCP no existe y la consulta falla.

## Cómo resolverlo rápido

---

Ejecuta la operación en la [documentación interactiva](#) con **Try it out**: ahí ves el esquema exacto del body, los parámetros obligatorios y un ejemplo válido. Cuando funciona ahí, replícalo en tu código y compara ambos requests.

### Un truco para acotar el problema

Envía el body **incompleto** a propósito. Si la API responde 400 con un JSON claro (“Debe especificar el tipo de DTE”, “Debe especificar el RUT del receptor”), las validaciones funcionan, y eso reduce el problema a un campo con **valor** inválido, no a un campo ausente.

---

Última actualización el 11/09/2026

Documentación » Solución de problemas » Recibo HTML o "Error interno" en vez de JSON

## Recibo HTML o "Error interno" en vez de JSON

Diagnóstico de las respuestas 500 con página HTML y de las respuestas que llegan sin cuerpo.

Anonymous · 11/09/2026

---

## Recibo HTML o “Error interno” en vez de JSON

### Síntoma: HTTP 500 con una página HTML

---

La respuesta llega con `content-type: text/html` y una página titulada “**Error interno**”, en lugar de un JSON. Las causas, en orden de frecuencia:

**1. La estructura del request no es la que espera el recurso.** Es la causa más común. Si el JSON pasa las validaciones básicas pero un campo va en el nivel equivocado o con un valor inválido, la operación puede fallar de forma no controlada. Revisa [Errores 400: estructura del request](#).

**2. El proxy no está bien configurado.** Revisa [Errores de proxy](#).

**3. Estás usando un recurso para algo que no hace.** Por ejemplo, los recursos del Portal MiPYME solo generan **borradores**: intentar emitir un documento real por esa vía no va a funcionar, y las lecturas del mismo producto seguirán respondiendo 200. Ese contraste —todas las lecturas bien, la escritura con 500— es la firma de este caso.

**4. Una intermitencia puntual del recurso.** Ocurre y la corregimos. Un caso real: el recurso de PDF de BHE recibidas devolvía 500 con la consulta correctamente formada, y se corrigió del lado de la plataforma.

### Síntoma: HTTP 404 con una página HTML

---

La ruta no existe. Revisa:

- **La URL base:** `https://app.apigateway.cl/api/v2/`, no `apigateway.cl` ni `www.apigateway.cl`.
- **Una barra final de más:** un `/` sobrante antes de los parámetros de query, como `.../225415/?fecha_emision=...`, hace que la ruta no calce.
- **El formato del período:** `202605`, no `2026-05`.
- **Que el recurso exista en la versión que usas:** los paths de `/api/v1` no existen en `/api/v2` y viceversa.

## Síntoma: la respuesta llega vacía aunque el Content-Length indique contenido

---

Se ha reportado en recursos de Legacy consultados de forma incorrecta. En los casos revisados, la causa fue la forma de consumir el recurso: método de autenticación equivocado o estructura del body que el recurso no espera. Contrasta tu request con el ejemplo de la [documentación de la API V1](#).

### Antes de reportarlo

---

Si tras revisar la documentación el error persiste, [genera un ticket](#) con el **cURL real** que estás enviando, la respuesta exacta y la fecha y hora de los intentos. Revisa [qué datos enviar](#).

---

Última actualización el 11/09/2026

Documentación » Solución de problemas » Errores 409, lentitud e intermitencias del SII

## Errores 409, lentitud e intermitencias del SII

Cómo distinguir una indisponibilidad real del SII de un problema en el request o en el proxy, y cómo saber si una consulta cancelada alcanzó a ejecutarse.

Anonymous · 11/09/2026

---

## Errores 409, lentitud e intermitencias del SII

API Gateway es una **pasarela**: cualquier intermitencia del SII llega hasta tu integración. Lo importante es saber cuándo el problema realmente es del SII, porque a menudo no lo es.

### El mensaje

---

```
{
  "code": 409,
  "detail": "Los servidores del SII no han respondido de manera satisfactoria, están con problemas de caída o de capacidad (HTTP 503). Se recomienda esperar y reintentar la consulta más tarde."
}
```

### Si es intermitente: espera y reintenta

---

Si el error aparece a veces y otras veces la consulta funciona, es una intermitencia real. Espera y reintenta más tarde. No hay una ventana horaria garantizada, aunque las horas de menor carga suelen dar mejores resultados.

Las mantenciones programadas del SII se publican en el [blog](#).

### Si es permanente: probablemente no es el SII

---

Este es el diagnóstico clave. Si un recurso devuelve **409 siempre**, en distintos días y horarios, mientras **otros recursos del mismo producto funcionan** con las mismas credenciales, el mismo token y el mismo proxy, la causa está en tu request.

Dos casos reales que se veían exactamente así:

**Filtros en el nivel equivocado.** Los filtros de los recursos del Portal MiPYME deben ir anidados dentro de un objeto `filtros`. Enviarlos al nivel raíz producía `409/503` de forma constante durante días.

**Método de autenticación equivocado.** El mismo recurso consultado con certificado digital, cuando la documentación indica RUT y clave de un usuario autorizado, devolvía 409/503 de forma reproducible.

Revisa además que el rango de fechas cubra **un mes como máximo** y que el valor de ESTADO esté en la lista válida del recurso.

### La prueba que separa un caso del otro

Si el **listado** falla con 409 pero la **descarga de PDF** del mismo módulo funciona, con las mismas credenciales y la misma configuración, el problema está en la petición del listado. Una caída del SII afectaría a ambos.

## Lentitud

---

Tres explicaciones posibles:

**Algunas operaciones son lentas por diseño.** Cada consulta implica iniciar sesión en el SII y navegar por su sitio: de 4 a 8 segundos es normal en operaciones autenticadas. La primera consulta de una serie es la más lenta porque abre la sesión; las siguientes la reutilizan.

**Intermitencia del SII.** Si un recurso que respondía en 2 o 3 segundos pasa a tardar 30, y otros funcionan bien, lo más probable es el SII.

**Tu proxy no responde.** El síntoma característico: consultas que quedan procesando **varios minutos**, con tiempos casi idénticos entre sí, y terminan en 400 o directamente sin respuesta. Revisa [Errores de proxy](#).

## Dimensiona bien tus timeouts

---

Un timeout de 30 segundos es insuficiente para operaciones que autentican en el SII. Si tu cliente aborta antes de que la API responda, verás un error de cancelación en tu lado mientras la petición sigue procesándose del nuestro, lo que dificulta el diagnóstico.

Pero dimensionar bien **no es poner el número más alto posible**. Una operación autenticada tarda de 4 a 8 segundos, y un margen holgado cubre de sobra cualquier lentitud real del SII. Si tu cliente ya espera **minutos** y aun así no alcanza, subir el timeout no resuelve nada: solo alarga la espera antes del mismo error. Un timeout de esa magnitud que no alcanza es un **síntoma**, no la causa, y lo que corresponde es buscar por qué la consulta no está terminando.

Compara el tiempo que mide tu cliente con el que registra el [dashboard de la conexión](#): si tu cliente cortó antes, la diferencia lo delata.

### Antes de subir el timeout, descarta el proxy

Subir el timeout solo ayuda cuando la API **responde**, y tarde. Si la petición queda colgada y **siempre** termina cancelada por tu cliente, con tiempos casi idénticos entre una y otra, eso no es lentitud: lo más probable es que la salida al SII no se esté completando. Un proxy inalcanzable no devuelve error, deja la consulta esperando. Revisa [Errores de proxy](#) antes de tocar el timeout.

## Si tu cliente canceló, ¿alcanzó a ejecutarse la operación?

Cuando tu cliente aborta te quedas sin respuesta, pero del lado nuestro la petición siguió su curso. Antes de reintentar —y con más razón si la operación **emite** un documento, porque un reintento a ciegas puede duplicarlo— compruébalo en el panel “**Actividad reciente (Últimas 10 peticiones)**” del [dashboard de la conexión](#), donde cada llamada queda con su endpoint, su código de respuesta, los créditos consumidos y el tiempo real que tomó. Míralo cuanto antes: solo conserva las diez peticiones más recientes.

| Lo que ves en el panel         | Qué significa                                                                          |
|--------------------------------|----------------------------------------------------------------------------------------|
| Aparece con un código de éxito | Se ejecutó. Confírmalo en el recurso de consulta del producto antes de volver a emitir |
| Aparece con un código de error | La operación falló. Puedes reintentar                                                  |
| No aparece                     | <b>No es concluyente.</b> Espera y vuelve a mirar antes de reintentar                  |

### Que no aparezca no significa que no haya llegado

El registro se escribe cuando la petición **termina**. Mientras sigue en curso —y una consulta detenida en un proxy que no responde puede seguir en curso varios minutos después de que tu cliente se rindió— no figura en el panel. La ausencia no distingue entre “nunca llegó” y “todavía se está ejecutando”. Espera a que cierre y vuelve a mirar.

## ¿Está caído el servicio?

Antes de concluirlo, descarta en orden:

1. **¿Es el día 1 del mes o pasó hace poco?** Puede ser una suspensión por créditos: la API responde `La conexión está inactiva`.
2. **¿Qué dice exactamente el error?** No todos son caídas: `401` es token o conexión, `403` productos, `402` créditos o IP, `400` con detalle de proxy es tu proxy.
3. **¿Falla un recurso o todos?** Si falla uno solo, no es una caída.
4. **¿Tu proxy sigue arriba y con la misma IP?**
5. **¿El SII está operativo?** Entra manualmente al portal con las mismas credenciales. Si tampoco funciona ahí, la falla es del SII.

---

Última actualización el 11/09/2026

Documentación » Solución de problemas » Errores 429 y bloqueos por exceso de uso

## Errores 429 y bloqueos por exceso de uso

Cómo funciona el límite de consultas, por qué se produce el bloqueo automático y cómo evitarlo.

Anonymous · 11/09/2026

---

# Errores 429 y bloqueos por exceso de uso

## El límite

---

El límite estándar es de **60 consultas por minuto** por conexión. Los límites de API Gateway están basados en los del SII y se ajustan si estos cambian.

Además, el **propio SII** limita la cantidad de consultas seguidas que recibe desde un mismo origen, y esas respuestas también llegan como 429.

## El bloqueo automático

---

El sistema aplica un **bloqueo automático** cuando una cuenta acumula demasiadas respuestas 429 en un mismo día.

Un caso real: entre las 11:42 y las 11:58 una cuenta recibió 11 respuestas 429 **provenientes del SII**, y con eso se superó el umbral y el bloqueo se aplicó.

## Qué lo provoca y cómo evitarlo

---

El patrón que provoca el bloqueo son las **ráfagas**. En el caso anterior las consultas salían hasta 8 en un mismo segundo.

- Envía las consultas **secuencialmente**, con una pausa breve entre cada una.
- Ante cualquier error, **espera unos segundos antes de reintentar**. No reintentes de inmediato.
- Lee **Retry-After** cuando venga en un 429 y respétala.
- Monitorea **X-RateLimit-Remaining** en las respuestas exitosas: si baja mucho, espacia las llamadas.
- **Cachea** en tu sistema los datos que no cambian a diario. Un caso real cachea la situación tributaria por 30 días y con eso su volumen real bajó a decenas de consultas al mes.

Las cabeceras disponibles:

| Cabecera              | Significado                            |
|-----------------------|----------------------------------------|
| X-RateLimit-Limit     | Tope de la ventana que aplica          |
| X-RateLimit-Remaining | Consultas que aún puedes hacer         |
| X-RateLimit-Reset     | Momento en que se renueva la ventana   |
| Retry-After           | Segundos sugeridos antes de reintentar |

## Volúmenes grandes

No existe un recurso que acepte **varios RUT en una misma solicitud**: cada consulta es individual. Para procesar miles de RUT, respeta el límite por minuto, procesa por lotes en ventanas de tiempo y cachea los resultados.

Considera también el costo total: cada consulta descuenta créditos, además de la base mensual del producto. Estímalo en el [cotizador](#), que considera la cantidad de consultas por recurso.

## No consultes rutas que no existen

Hacer muchas peticiones a rutas inexistentes —probando “a ver si responden”— puede provocar un **bloqueo de IP**. Si un recurso no aparece en la [documentación de la API](#), no existe: no hay recursos ocultos.

## Si crees que el bloqueo no corresponde

Puede ocurrir. Ha habido casos en que el bloqueo se aplicó por una intermitencia del propio sistema de bloqueo, o porque respuestas 429 provenientes del SII se contabilizaron como exceso de cuota propia. Esos casos se corrigen.

[Genera un ticket](#) indicando la fecha y hora del bloqueo, y adjunta tu registro de consumo.

## Si tu servicio es Legacy: la cuota es cada 24 horas

En API Gateway Legacy la cuota del plan **no se reinicia a medianoche**: es una **ventana móvil de 24 horas** desde tu primera consulta. Si tu primera consulta fue un lunes a las 11:00, la cuota se renueva el martes a las 11:00.

Esto explica que los gráficos de consumo por día no calcen con el momento del bloqueo. Un caso real vio 603 consultas en el gráfico del día y sin embargo se bloqueó, porque la ventana abarcaba consultas de la tarde anterior.

---

Última actualización el 11/09/2026