← Resumen

Configuración de OmniSCP

Todos los ajustes ajustables por el operador se proporcionan a través de `config/runtime.exs`, cada uno respaldado por una variable de entorno para que los valores puedan ser sobrescritos en el momento de la implementación sin editar el archivo. Los valores que se muestran a continuación son los valores predeterminados aplicados cuando la variable de entorno correspondiente no está establecida.

```
import Config

config :omniscp,
  sbi_scheme: System.get_env("SBI_SCHEME", "http"),
  sbi_addr: System.get_env("SBI_ADDR", "127.0.0.17"),
  sbi_port: String.to_integer(System.get_env("SBI_PORT", "7777")),
  nrf_uri: System.get_env("NRF_URI", "http://127.0.0.1:7777"),
  mcc: System.get_env("MCC", "999"),
  mnc: System.get_env("MNC", "70"),
  heartbeat_interval:
String.to_integer(System.get_env("HEARTBEAT_INTERVAL", "10000")),
  prometheus_metrics_port:
String.to_integer(System.get_env("PROMETHEUS_PORT", "9568")),
  discovery_cache_ttl:
String.to_integer(System.get_env("DISCOVERY_CACHE_TTL", "60000")),
  lb_strategy: String.to_atom(System.get_env("LB_STRATEGY",
"round_robin")),
  max_retries: String.to_integer(System.get_env("MAX_RETRIES",
"2")),
  upstream_timeout:
String.to_integer(System.get_env("UPSTREAM_TIMEOUT", "5000"))

config :logger, :default_formatter,
  format: {OmniLogger.JsonFormatter, :format},
  metadata: :all
```

Parámetros de `:omniscp`

Parámetro	Tipo	Requerido	Predeterminado
`sbi_scheme`	String	No	`"http"`
`sbi_addr`	String	No	`"127.0.0.17"`
`sbi_port`	Integer	No	`7777`
`sbi_certfile`	String	No	*(no establecido)*

Parámetro	Tipo	Requerido	Predeterminado
`sbi_keyfile`	String	No	*(no establecido)*
`nrf_uri`	String	No	`"http://127.0.0.1:777`
`mcc`	String	No	`"999"`
`mnc`	String	No	`"70"`

Parámetro	Tipo	Requerido	Predeterminado
`heartbeat_interval`	Integer (ms)	No	`10000`
`prometheus_metrics_port`	Integer	No	`9568`
`discovery_cache_ttl`	Integer (ms)	No	`60000`

Parámetro	Tipo	Requerido	Predeterminado
`lb_strategy`	Atom	No	`round_robin`
`max_retries`	Integer	No	`2`

Parámetro	Tipo	Requerido	Predeterminado
`upstream_timeout`	Integer (ms)	No	`5000`

Capacidad/prioridad del perfil NRF

Estos dos valores son parte del propio perfil NRF del SCP. **No** se establecen en `runtime.exs` y no tienen una variable de entorno dedicada, por lo que normalmente toman sus valores predeterminados, pero son ajustables en

tiempo de ejecución a través de la API de gestión (`PATCH /api/oam/config/{id}`), lo que también desencadena una re-registro en el NRF para que el nuevo valor sea publicitado.

Parámetro	Tipo	Requerido	Predeterminado	Var. de Entorno	
`nf_capacity`	Integer (0-65535)	No	`100`	(ninguna, solo OAM)	Val pub per Los con pue por
`nf_priority`	Integer (0-65535)	No	`0`	(ninguna, solo OAM)	Val pub per (me pric

> **Valores predeterminados de respaldo compilados.** La aplicación también lleva valores predeterminados de respaldo compilados que se utilizan solo si la clave `:omniscp` está completamente ausente, y estos difieren de los valores predeterminados del entorno `runtime.exs` anteriores (por ejemplo, `sbi_addr` vuelve a `127.0.0.200`, `nrf_uri` a `http://127.0.0.10:7777`, y `max_retries` a `1`). En operación normal, `runtime.exs` proporciona cada clave, por lo que los valores en las tablas anteriores son los valores predeterminados efectivos.

Ajustes de balanceador de carga y sobrecarga

Estas claves de configuración de la aplicación `:omniscp` rigen el seguimiento de salud basado en fallos automático y el control de sobrecarga. **No tienen variable de entorno** y no están expuestas a través de la API OAM, por lo que se establecen solo a través de la configuración de la aplicación. Cada una se lee en tiempo real durante la ejecución, por lo que un cambio en la

configuración de la aplicación en ejecución entra en vigor en la siguiente solicitud sin necesidad de recompilación. Los valores predeterminados mostrados se aplican cuando la clave no está establecida.

Parámetro	Tipo	Requerido	Predeterminado	
`health_check_interval`	Integer (ms)	No	`15000`	Int ba sa en so ins pr
`failure_threshold`	Integer	No	`3`	Fa co an ins ma sa eli rot
`recovery_cooldown_ms`	Integer (ms)	No	`30000`	Tie de un sa ele vo rot
`oci_reduction_threshold`	Integer	No	`50`	Ur de rig pr so (se de `3g` eli co

Parámetro	Tipo	Requerido	Predeterminado	
				ca de ba ca Op En De

Ajuste del oyente SBI

El proxy SBI se sirve a través de HTTP/2. Los siguientes parámetros de transporte están fijos en la aplicación (no ajustables por el entorno) y se documentan aquí porque rigen el comportamiento de concurrencia del SCP bajo carga:

Configuración	Valor	Descripción
`max_concurrent_streams`	`5000`	Máximo de flujos HTTP/2 concurrentes por conexión entrante. Limita cuántas solicitudes SBI en vuelo puede multiplexar una única conexión de consumidor a través del proxy.
`num_acceptors`	`100`	Tamaño del grupo de aceptadores del oyente SBI para aceptar nuevas conexiones TCP entrantes.

Las solicitudes salientes (upstream) a los productores utilizan un grupo de conexiones compartido de Finch proporcionado por el runtime común de Omni5G, reutilizado en todo el tráfico reenviado y indexado por el host del productor. El tamaño de su grupo no se expone como un ajuste de OmniSCP, por lo que la concurrencia upstream está gobernada por ese grupo compartido junto con `upstream_timeout` (que limita cada solicitud reenviada individual) y el límite de `max_concurrent_streams` entrante mencionado anteriormente.

Para escalar un SCP ocupado, los controles que puedes manejar son `upstream_timeout` (tiempos de espera más cortos liberan conexiones de vuelta al grupo más pronto) y `discovery_cache_ttl` (una mayor tasa de aciertos elimina los viajes de ida y vuelta al NRF del camino de solicitud); `max_concurrent_streams` limita cuánto puede multiplexar una única conexión de consumidor a través del proxy.

API de Gestión / OAM

OmniSCP expone una API de gestión/OAM a través de **HTTPS en el puerto 8443** para el estado de solo lectura y control operativo en vivo. Estos puntos finales no llevan tráfico de suscriptores; restringe el acceso a la red de operaciones. Todos los caminos a continuación se sirven bajo la ruta base `/api` del oyente de gestión.

Puntos finales de estado (solo lectura)

Método	Ruta	Descripción
`GET`	`/api/status/api`	Estado del servicio API.
`GET`	`/api/status/license`	Estado de la licencia.
`GET`	`/api/status/nrf`	Estado de registro NRF para el SCP.
`GET`	`/api/status/nf`	Estado del NF SCP: dirección/puerto SBI, tamaño de caché de descubrimiento, estrategia `lb_strategy` activa y salud del balanceador de carga por instancia.
`GET`	`/api/api/cache`	Contenidos de la caché de descubrimiento: por entrada el tipo de NF objetivo, nombre del servicio, conteo de instancias y TTL restante.
`GET`	`/api/api/health`	Salud del balanceador de carga: por instancia rastreada el indicador de salud y conteo de fallos, más totales saludables/no saludables.
`GET`	`/api/api/statistics`	Estadísticas del SCP: tamaño de la caché de descubrimiento y conteos de instancias rastreadas/saludables/no saludables.
`GET`	`/api/oam/config`	Configuración de tiempo de ejecución efectiva actual (SBI,

Método	Ruta	Descripción
		NRF, TTL de caché, `lb_strategy`, `max_retries`, `upstream_timeout`, `nf_capacity`, `nf_priority`, nivel de registro).
`GET`	`/api/oam/cache`	Vista de la caché de descubrimiento (como `/api/api/cache`).
`GET`	`/api/oam/upstream/drain`	Lista las instancias de NF upstream actualmente drenadas.

Puntos finales de control

Método	Ruta	Descripción
`PATCH`	`/api/oam/config/{id}`	Actualizar configuraciones de tiempo de ejecución seleccionadas en vivo: `discovery_cache_ttl`, `lb_strategy`, `max_retries`, `upstream_timeout`, `nf_capacity`, `nf_priority`, `log_level`. Los valores son validados (`lb_strategy` debe ser uno de `round_robin`, `weighted`, `priority`; las claves enteras deben ser enteros). Cambiar `nf_capacity`/`nf_priority` también desencadena una re-registro en el NRF.
`DELETE`	`/api/oam/cache/{id}`	Vaciar la caché de descubrimiento para un tipo de NF específico, o pasar `all` para vaciar toda la caché.
`POST`	`/api/oam/upstream/drain`	Drenar un upstream para que el proxy deje de seleccionarlo. Cuerpo: `{"nf_instance_id": "..."}` (alias `"target"`), opcional `"reason"`.
`DELETE`	`/api/oam/upstream/drain/{id}`	Desdrainar un upstream por ID de instancia de NF,

Método	Ruta	Descripción
		haciéndolo elegible para selección nuevamente.
`POST`	`/api/oam/log_level`	Cambiar el nivel de registro activo en tiempo de ejecución. Cuerpo: `{"level": "..."}` (`emergency`, `alert`, `critical`, `error`, `warning`/`warn`, `notice`, `info`, `debug`).
`POST`	`/api/oam/nrf/reregister`	Forzar una re-registro inmediata del SCP con el NRF.

Drenaje upstream es un estado administrativo, separado del seguimiento de salud basado en fallos automático. Las instancias drenadas se filtran de la lista de candidatos *antes* de la selección del balanceador de carga, por lo que un productor drenado nunca recibe tráfico reenviado. Si **cada** candidato para un servicio está drenado, el proxy no tiene un siguiente salto utilizable y las solicitudes a ese servicio devuelven `504 NF_DISCOVERY_FAILURE` (ver Operaciones → Respuestas de error del proxy y Solución de problemas).

Registro

Parámetro	Tipo	Requerido	Predeterminado	Descrip
`format`	Tuple	No	`{OmniLogger.JsonFormatter, :format}`	Formate líneas d registro como JS estructu a través formate compar `OmniLog` adecua para la ingestió una tub de regis
`metadata`	Atom / List	No	`:all`	Metada de regis emitir. incluye clave d metada adjunta una ent de regis

El nivel de registro activo puede cambiarse en tiempo de ejecución sin reiniciar a través de la API de gestión, ya sea `POST /api/oam/log_level` o `PATCH /api/oam/config/{id}` con un valor de `log_level`.

← Resumen

Métricas de OmniSCP

Las métricas se exponen en el listener de Prometheus (`prometheus_metrics_port`, por defecto `9568`) en `/metrics`. Los valores de etiquetas de resultado en las métricas de proxy son: `success` (2xx/3xx), `client_error` (4xx), `server_error` (5xx), `error` (conexión/tiempo de espera).

Métricas de proxy

Métrica: `omni_scp_proxy_requests_count`
Tipo: Contador
Descripción: Conteo del resultado del proxy por solicitud.
Etiquetas: `target_nf_type`, `result`

Métrica: `omni_scp_proxy_requests_duration_ms`
Tipo: Resumen
Descripción: Duración del proxy por solicitud en milisegundos.
Etiquetas: `target_nf_type`

Métrica: `omni_scp_proxy_requests_total`
Tipo: Contador
Descripción: Total de solicitudes proxy por tipo de NF objetivo y resultado.
Etiquetas: `target_nf_type`, `result`

Métrica: `omni_scp_proxy_request_duration_ms`
Tipo: Histograma (distribución)
Descripción: Duración de la solicitud proxy. Cubos: 1, 5, 10, 25, 50, 100, 250, 500, 1000 ms.
Etiquetas: `target_nf_type`

El SCP es un proxy de reenvío sin estado y no mantiene estado por asociación, por lo que no expone ninguna métrica de "asociaciones activas". La actividad en vivo se observa a través del contador de solicitudes proxy anterior (tasa,

ratio de errores) y las cifras de salud del balanceador de carga en la API de OAM.

```
# Tasa de solicitudes proxy por tipo de NF objetivo
sum by (target_nf_type) (rate(omni_scp_proxy_requests_total[5m]))

# Ratio de errores del servidor entre todos los productores
sum(rate(omni_scp_proxy_requests_total{result="server_error"}
[5m]))
  / sum(rate(omni_scp_proxy_requests_total[5m]))

# Latencia p95 del proxy por tipo de NF
histogram_quantile(0.95, sum by (le, target_nf_type)
(rate(omni_scp_proxy_request_duration_ms_bucket[5m])))
```

Métricas de caché de descubrimiento

Métrica: `omni_scp_discovery_cache_hits`
Tipo: Contador
Descripción: Aciertos en la caché de descubrimiento, por servicio.
Etiquetas: `target_nf_type`, `service_name`

Métrica: `omni_scp_discovery_cache_misses`
Tipo: Contador
Descripción: Fallos en la caché de descubrimiento, por servicio.
Etiquetas: `target_nf_type`, `service_name`

Métrica: `omni_scp_cache_hits_total`
Tipo: Contador
Descripción: Aciertos agregados en la caché de descubrimiento.
Etiquetas: ninguna

Métrica: `omni_scp_cache_misses_total`
Tipo: Contador
Descripción: Fallos agregados en la caché de descubrimiento.
Etiquetas: ninguna

```
# Ratio general de aciertos en la caché (objetivo > 0.9 en una
topología estable)
sum(rate(omni_scp_cache_hits_total[5m]))
  / (sum(rate(omni_scp_cache_hits_total[5m])) +
sum(rate(omni_scp_cache_misses_total[5m])))

# Tasa de fallos por servicio (valores altos significan
redescubrimiento frecuente de NRF)
sum by (target_nf_type, service_name)
(rate(omni_scp_discovery_cache_misses[5m]))
```

Métrica de registro de NRF

Métrica: `omni_scp_nrf_registration_status`
Tipo: Gauge
Descripción: Estado de registro del SCP con el NRF (`1` = registrado, `0` = no registrado).
Etiquetas: `nf_type`

```
# Alerta si el SCP pierde su registro en el NRF
min(omni_scp_nrf_registration_status) < 1
```

Métricas de BEAM VM

El runtime también exporta gauges de host/VM: `beam_memory_total`, `beam_memory_processes`, `beam_memory_processes_used`, `beam_memory_system`, `beam_memory_atom`, `beam_memory_atom_used`, `beam_memory_binary`, `beam_memory_code`, `beam_memory_ets` (todos en bytes), y `beam_processes_count`, `beam_ports_count`, `beam_atom_count`, `beam_vm_uptime` (segundos). Estos son útiles para el monitoreo de capacidad y fugas del proceso SCP en sí.

> Los nombres de las métricas anteriores utilizan la normalización de punto a guion bajo de Prometheus; las series de histograma exponen los sufijos habituales `_bucket`, `_sum` y `_count`.

← Resumen

Operaciones de OmniSCP

Esta guía cubre el papel del SCP en 3GPP, las interfaces y puntos finales que expone, su comportamiento de enrutamiento de solicitudes y descubrimiento, y los procedimientos clave en tiempo de ejecución con diagramas de secuencia. Para cada configuración ajustable, consulte Configuración; para métricas, consulte Métricas; para el manejo de fallos, consulte Solución de problemas.

Papel de 3GPP y referencias de especificación

El SCP proporciona **comunicación indirecta** según lo definido en TS 23.501. En **Modelo C**, el consumidor realiza el descubrimiento por sí mismo (a través del NRF) pero delega el reenvío de solicitudes al SCP. En **Modelo D**, el consumidor delega tanto el descubrimiento como el reenvío al SCP: envía la solicitud con parámetros `3gpp-Sbi-Discovery-*` y el SCP realiza el descubrimiento NRF en su nombre. OmniSCP admite ambos modelos, además del reenvío directo cuando el consumidor ya conoce el objetivo.

Especificación	Sección	Relevancia
TS 23.501	§6.3.1, Anexo E	Papel del SCP y los modelos de comunicación indirecta (C y D) en el 5G SBA.
TS 23.501	§7.3	Comportamiento de manejo de solicitudes del SCP dentro del SBA.
TS 29.500	§6.10	Enrutamiento SBI a través del SCP; los encabezados de enrutamiento/descubrimiento `3gpp-Sbi-*`.
TS 29.500	§6.10.3	Conjunto de encabezados `3gpp-Sbi-Discovery-*` transportados en solicitudes de descubrimiento delegadas.
TS 29.500	§6.10.3.2	Encabezado `3gpp-Sbi-Target-apiRoot` (objetivo de enrutamiento directo); selección del nombre del servicio principal.
TS 29.500	§6.10.3.3	Encabezado de respuesta `3gpp-Sbi-Producer-Id` (vinculación de consumidor a productor).
TS 29.500	§5.2.2.2	El encabezado `User-Agent` transmite el tipo de NF del solicitante.
TS 29.510	§6.2	Nnrf_NFDiscovery, el servicio de descubrimiento NRF que consume el SCP.
TS 29.510	§6.1	Nnrf_NFManagement, auto-registro del SCP y notificaciones de estado de NF.

Referencia: TS 23.501, TS 29.500, TS 29.510.

Interfaces y oyentes

OmniSCP es un proxy inverso HTTP de propósito general. Los consumidores de NF dirigen su tráfico SBI al oyente SBI del SCP en lugar de hacerlo directamente a los productores. Para cada solicitud, el SCP resuelve un productor objetivo (a partir de un encabezado de solicitud, de un resultado de descubrimiento NRF en caché, o consultando el NRF), selecciona una instancia a través de la estrategia de balanceo de carga configurada, reenvía la solicitud y, en caso de fallo, vuelve a intentar contra una instancia diferente.

Oyente	Puerto predeterminado	Esquema	Propó
Proxy SBI	`7777` (`sbi_port`)	HTTP / HTTPS (`sbi_scheme`)	Acepta solicituc consumidores y notificaciones c NRF. Todos los c no son de notifi envían a los prc Establezca `sbi_scheme=ht` `sbi_certfile`/ para terminar T oyente.
Métricas de Prometheus	`9568` (`prometheus_metrics_port`)	HTTP	Sirve el punto f raspado `/metri` Consulte Métric
API de gestión / OAM	`8443`	HTTPS	Puntos finales c solo lectura y c Consulte Config API de gestión.

El oyente SBI se sirve a través de HTTP/2 con un límite de flujo por conexión y un grupo de aceptadores fijo; esos valores de ajuste de transporte se describen en Configuración → Ajuste del oyente SBI.

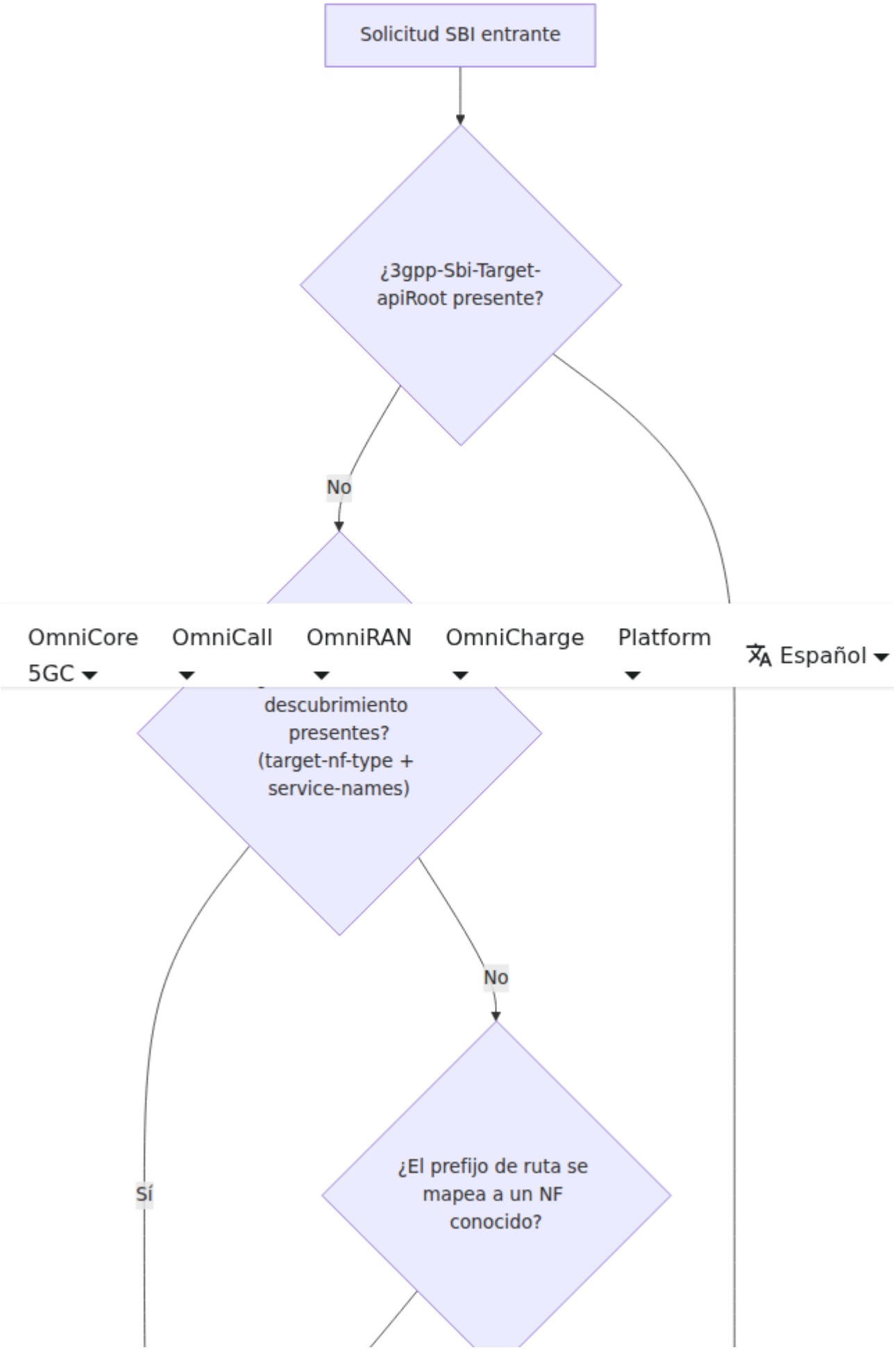
Tabla de puntos finales SBI

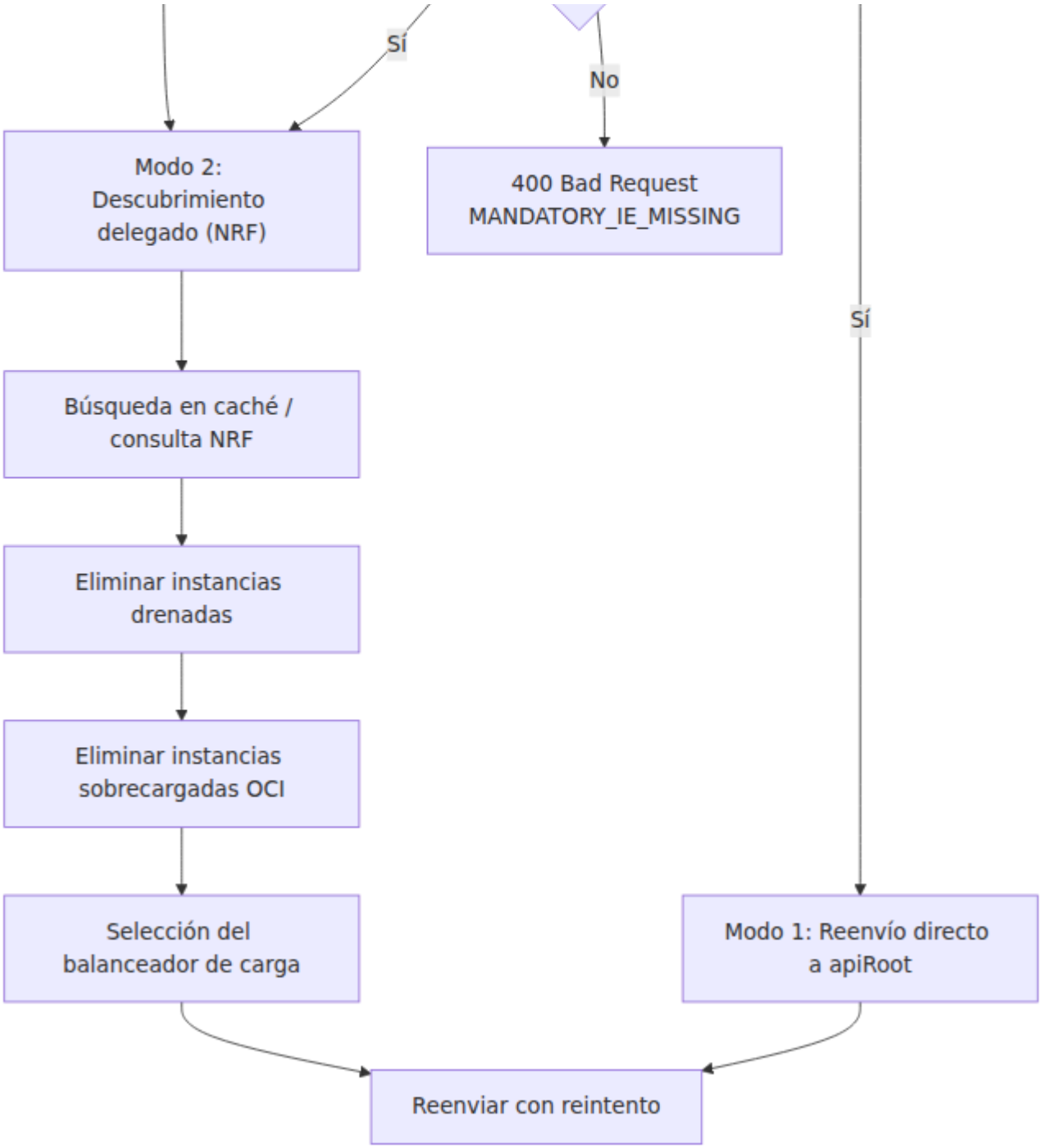
OmniSCP es transparente: exactamente una ruta se maneja localmente, y cada otra solicitud se envía a través del proxy.

Método	Ruta	Manejado localmente	Descripción
`POST`	`/nnrf-nfm/v1/nf-status-notify`	Sí	Recibe notificaciones de estado de NF del NRF. En `NF_DEREGISTERED` o `NF_PROFILE_CHANGED`, la caché de descubrimiento se invalida para la única instancia afectada identificada por su `nfInstanceId` (volviendo a la eliminación de un tipo de NF solo si no hay id de instancia presente, y no haciendo nada si ninguno está presente). Nunca se borra toda la caché. Devuelve `204 No Content`.
`*`	todas las demás rutas	No (enviado a través del proxy)	Reenviado al productor resuelto según el modo de enrutamiento activo.

Enrutamiento y descubrimiento

En cada solicitud entrante, el SCP inspecciona los encabezados de la solicitud y la ruta para decidir **cómo** resolver el productor objetivo. Se evalúan tres modos de enrutamiento en orden de prioridad:


Solicitud SBI entrante
¿3gpp-Sbi-Target-apiRoot presente?
No
OmniCore 5GC
OmniCall
OmniRAN
OmniCharge
Platform
Español
descubrimiento presentes? (target-nf-type + service-names)
No
Sí
¿El prefijo de ruta se mapea a un NF conocido?



Modo 1: Reenvío directo

Cuando el consumidor envía `3gpp-Sbi-Target-apiRoot`, ya conoce la raíz de la API del productor. El SCP elimina los encabezados específicos del SCP y reenvía la solicitud directamente a esa URI base sin búsqueda de NRF, caché o balanceo de carga.

Modo 2: Descubrimiento delegado (Modelos C/D)

Cuando el consumidor envía `3gpp-Sbi-Discovery-target-nf-type` y `3gpp-Sbi-Discovery-service-names`, el SCP resuelve el conjunto de productores por sí mismo:

1. **Búsqueda en caché**: indexada por `{tipo de NF objetivo, nombre de servicio principal}`. El primer nombre en `3gpp-Sbi-Discovery-service-names` se trata como el servicio principal (TS 29.500 §6.10.3.2). En un acierto dentro del TTL, se utiliza la lista de instancias en caché y no se realiza ninguna consulta NRF.
2. **Descubrimiento NRF**: en un fallo, el SCP consulta el NRF (Nnrf_NFDiscovery), pasando los parámetros de descubrimiento que recibió: tipo de NF solicitante, lista de PLMN objetivo, lista de S-NSSAI solicitante, ID del conjunto de NF y ID de instancia de NF objetivo. El resultado se almacena en caché bajo la misma clave durante `discovery_cache_ttl`.
3. **Filtrado de drenaje**: cualquier instancia cuyo `nfInstanceId` esté drenado administrativamente (consulte Configuración → API de gestión) se elimina

del conjunto de candidatos antes de la selección.

4. **Control de sobrecarga**: cualquier instancia que actualmente esté señalando sobrecarga a través de su encabezado de respuesta `3gpp-Sbi-Oci` (Información de Control de Sobrecarga) en o por encima de `oci_reduction_threshold` durante su período de validez se elimina del conjunto de candidatos (TS 29.500 §6.3). Esto es un esfuerzo de buena fe: si todos los candidatos están sobrecargados, se mantiene el conjunto completo en lugar de rechazar la solicitud.
5. **Selección del balanceador de carga**: se elige una instancia de los candidatos restantes utilizando `lb_strategy`.
6. **Reenviar con reintento**: la solicitud se reenvía; en caso de fallo, se prueba una instancia diferente, hasta `max_retries`. En cada respuesta, se registra el `3gpp-Sbi-Oci` anunciado por el productor para que las solicitudes posteriores puedan reducir la carga de él.

Modo 3: Inferencia basada en la ruta

Cuando no hay encabezados de enrutamiento presentes, el SCP infiere el tipo de NF objetivo a partir del prefijo del nombre del servicio de la ruta de la solicitud (formato de ruta `/<nombre-del-servicio>/<versión>/...`) y luego procede exactamente como en el Modo 2.

Prefijo de ruta	Tipo de NF
`nudm-`	UDM
`nausf-`	AUSF
`namf-`	AMF
`nsmf-`	SMF
`npcf-`	PCF
`nudr-`	UDR
`nnssf-`	NSSF
`nbsf-`	BSF
`nnrf-`	NRF
`nchf-`	CHF
`nnef-`	NEF
`naf-`	AF

Si el prefijo de la ruta no está en esta tabla, el SCP devuelve `400 Bad Request` con causa `MANDATORY_IE_MISSING`.

Estrategias de balanceo de carga

La estrategia de selección se establece mediante `lb_strategy` y se aplica a los Modos 2/3 (conjuntos de instancias descubiertas):

Estrategia	Lógica de selección
`round_robin`	Cicla a través de las instancias candidatas saludables en orden. Se mantiene un contador de rotación separado por grupo `{nfType, serviceName}` para que diferentes servicios no interfieran. Este es el valor predeterminado.
`weighted`	Selecciona la instancia con la mejor (más baja) puntuación de `load - capacity` de su perfil NRF, prefiriendo productores de alta capacidad y baja carga. Si falta `load`, se usa 50 y si falta `capacity`, se usa 100.
`priority`	Selecciona la instancia con el valor de `priority` más bajo (mayor prioridad) de su perfil NRF. Si falta `priority`, se usa 65535. Adecuado para implementaciones activas/en espera.

> **De dónde provienen los pesos.** Las estrategias `weighted` y `priority` leen `load`, `capacity` y `priority` de **cada perfil NRF de productor**; el SCP no tiene una configuración de ponderación por productor propia. Para sesgar la selección, ajuste esos valores en los productores (o en el NRF), no en el SCP. Los ajustes `nf_capacity`/`nf_priority` del SCP (Configuración → Capacidad/prioridad del perfil NRF) son un concepto separado: describen *el SCP en sí* en su propio perfil para que los pares puedan ponderar el SCP, y no afectan cómo el SCP pondera a los productores.

Seguimiento de salud basado en fallos. Independientemente de la estrategia, el balanceador de carga rastrea los resultados por instancia. Una instancia se marca como no saludable después de **3 fallos consecutivos** y se excluye de la selección; una barrido de salud en segundo plano se ejecuta cada **15 segundos** y devuelve una instancia al servicio una vez que ha transcurrido un **enfriamiento de 30 segundos** desde su último fallo. Si todos los candidatos están no saludables, el SCP vuelve a la lista completa en lugar de fallar directamente (priorizando la disponibilidad sobre la exclusión estricta). Este seguimiento automático de salud es distinto del estado de drenaje impulsado por el operador.

Estos tres umbrales (conteo de fallos, intervalo de barrido y enfriamiento) no se exponen como variables de entorno y tienen los valores predeterminados como se muestra arriba, pero son claves de configuración de aplicación reales `:omniscp` (`failure_threshold`, `health_check_interval`, `recovery_cooldown_ms`) que se leen en vivo en tiempo de ejecución. Ningún punto final de OAM los cambia en tiempo de ejecución. Consulte Configuración → Ajustes de balanceador de carga y sobrecarga. Para sacar un productor de la rotación en un horario que controle (en lugar de esperar el umbral de fallo), utilice drenaje administrativo en su lugar. Un fallo para este propósito es una respuesta 5xx o un error de conexión/timeout; cualquier respuesta con estado por debajo de 500 (incluyendo 4xx) cuenta como un éxito y restablece el contador de fallos de la instancia.

Comportamiento del SBI / Proxy

Encabezados consumidos (y eliminados antes del reenvío)

Estos controlan el enrutamiento y nunca se envían al productor.

Encabezado	Propósito
`3gpp-Sbi-Target-apiRoot`	Objetivo de enrutamiento directo del Modo 1.
`3gpp-Sbi-Discovery-target-nf-type`	Tipo de NF a descubrir (por ejemplo, `UDM`).
`3gpp-Sbi-Discovery-service-names`	Nombres de servicios separados por comas; el primero es el principal (utilizado como clave de caché y servicio de consulta NRF).
`3gpp-Sbi-Discovery-requester-nf-type`	Tipo de NF solicitante; limita la consulta NRF.
`3gpp-Sbi-Discovery-target-plmn-list`	Lista de PLMN objetivo; pasada a través de la consulta NRF.
`3gpp-Sbi-Discovery-requester-snssai-list`	Lista de S-NSSAI solicitante; pasada a través de la consulta NRF.
`3gpp-Sbi-Discovery-nf-set-id`	Filtro de ID de conjunto de NF; pasado a través de la consulta NRF.
`3gpp-Sbi-Discovery-target-nf-instance-id`	ID de instancia de NF objetivo específica; pasada a través de la consulta NRF.
`3gpp-Sbi-Discovery-requester-nf-instance-id`	ID de instancia solicitante. Consumido y eliminado, pero no actualmente incluido en la consulta de descubrimiento NRF.

El encabezado `User-Agent` también se inspecciona: según TS 29.500 §5.2.2.2, comienza con el tipo de NF del solicitante (por ejemplo, `AMF-...`), que se utiliza como tipo de NF solicitante si ningún encabezado de descubrimiento

explícito proporciona uno. Si ninguno está presente, el SCP predetermina el tipo de NF solicitante a `SCP` para la consulta NRF.

Encabezados producidos

Encabezado	Propósito
`3gpp-Sbi-Producer-Id`	Agregado a la respuesta, transportando el `nfInstanceId` del productor que atendió la solicitud, para que el consumidor pueda vincular solicitudes posteriores al mismo productor (TS 29.500 §6.10.3.3).
`3gpp-Sbi-Lci`	Información de Control de Carga del Productor, retransmitida de extremo a extremo desde la respuesta del productor para que el consumidor reciba su indicación de carga a través del SCP (TS 29.500 §6.3). El SCP retransmite el valor del productor y no sintetiza el suyo propio.
`3gpp-Sbi-Load`	El encabezado compañero de carga `3gpp-Sbi-Lci`, retransmitido de extremo a extremo desde la respuesta del productor junto con `3gpp-Sbi-Lci` (TS 29.500 §6.3).

Los encabezados de respuesta por salto (`transfer-encoding`, `connection`, `content-length`) no se reenvían de vuelta al consumidor.

Respuestas de error del proxy

Cuando el SCP no puede completar una operación, devuelve un cuerpo `ProblemDetails` RFC 7807 según TS 29.500.

Estado HTTP	Causa	Condición
`400 Bad Request`	`MANDATORY_IE_MISSING`	Sin información de enrutamiento: sin target-apiRoot, sin encabezados de descubrimiento, y el prefijo de ruta no se mapea a ningún servicio conocido.
`502 Bad Gateway`	`TARGET_NF_NOT_REACHABLE`	No se pudo resolver ninguna URI SBI de un perfil de NF descubierto.
`504 Gateway Timeout`	`NF_DISCOVERY_FAILURE`	El descubrimiento (NRF + caché) no produjo instancias de NF utilizables, incluyendo cuando **cada** instancia candidata para el servicio está drenada administrativamente.
`500 Internal Server Error`	`SYSTEM_FAILURE`	Error interno del proxy genuinamente inesperado que no está cubierto por las causas más específicas anteriores.

> **Nota sobre el caso de todos drenados.** Si cada candidato para un servicio está drenado, el proxy no tiene un siguiente salto utilizable y devuelve `504 NF_DISCOVERY_FAILURE` (no `500`). Drene al menos una instancia para restaurar el servicio. Consulte Solución de problemas.

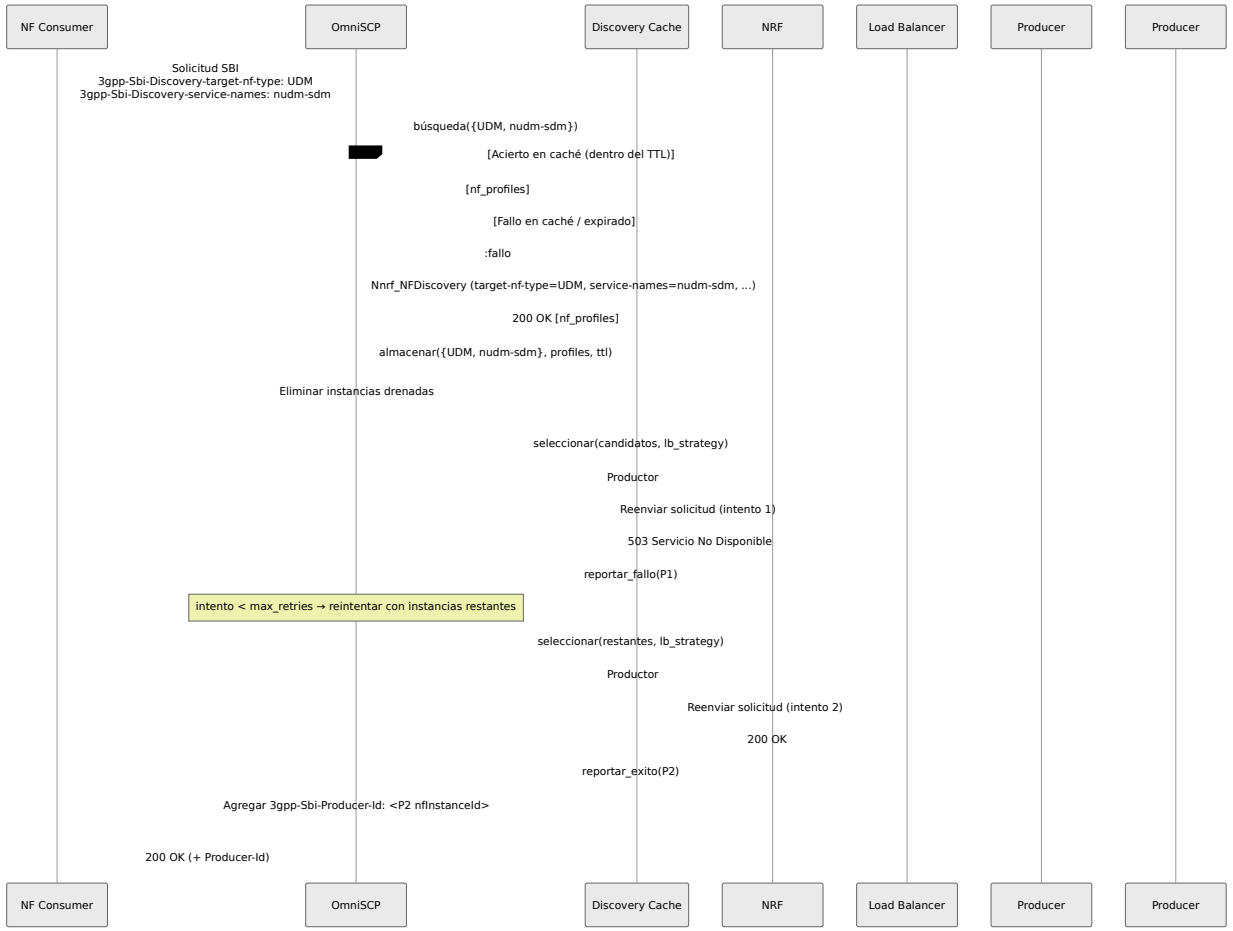
Semántica de reintentos. Una respuesta de productor con estado **< 500** (incluyendo 4xx) se trata como una respuesta válida y se devuelve al consumidor tal cual. Una respuesta **5xx** o un **error de conexión/timeout** cuenta como un fallo: la instancia fallida se informa al balanceador de carga y se elimina del conjunto de candidatos, y se prueba una instancia diferente. Los reintentos continúan hasta que se obtiene una respuesta no fallida, se alcanza

`max_retries`, o se agota la lista de candidatos, momento en el cual se devuelve la última respuesta (o error).

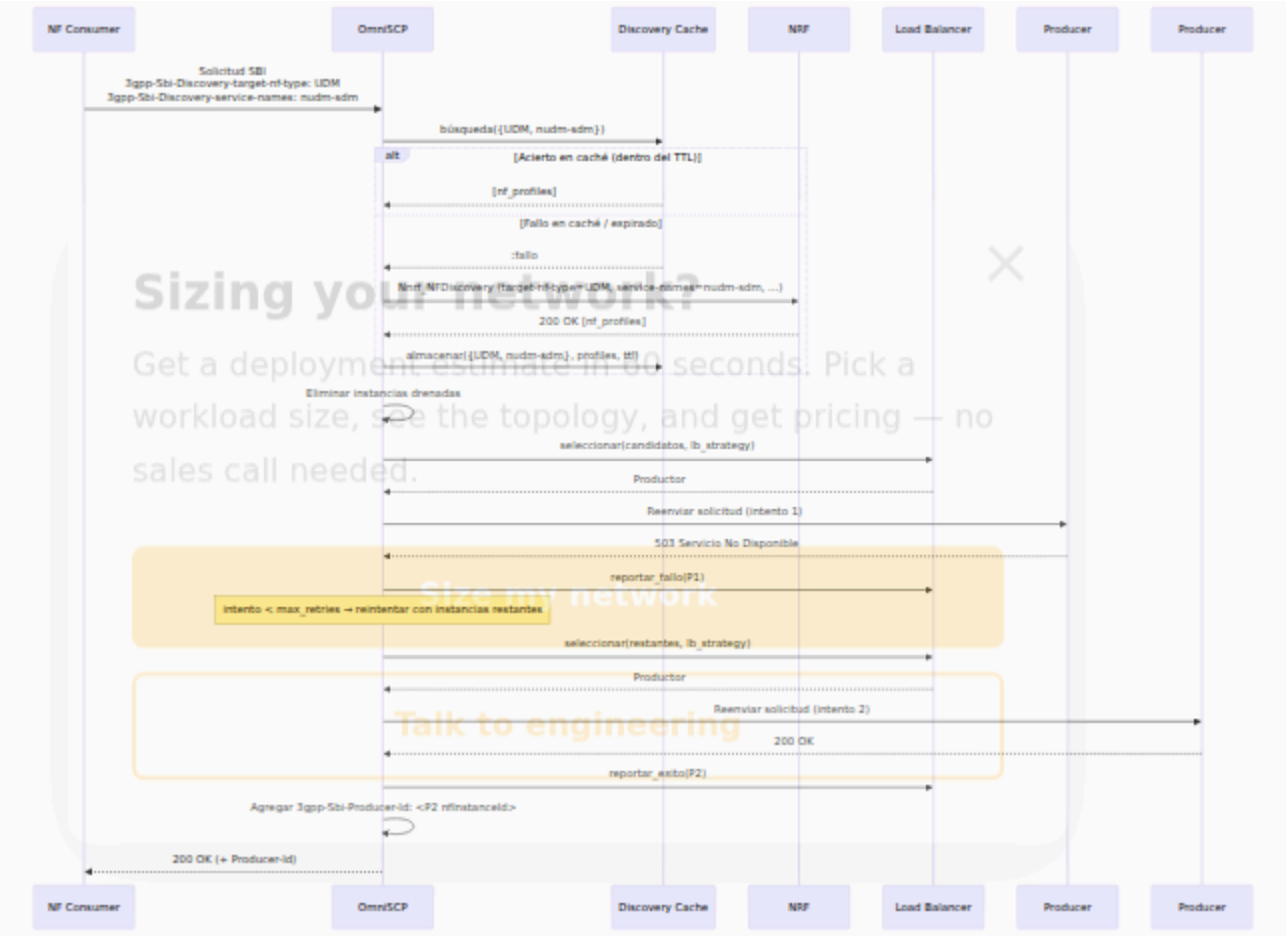
Procedimientos Clave

Solicitud enviada a través del proxy, balanceada con un reintento (Modo 2/3)

La secuencia a continuación muestra el descubrimiento delegado, la selección del balanceador de carga, un primer intento que falla con un 5xx, y un reintento exitoso contra una segunda instancia.

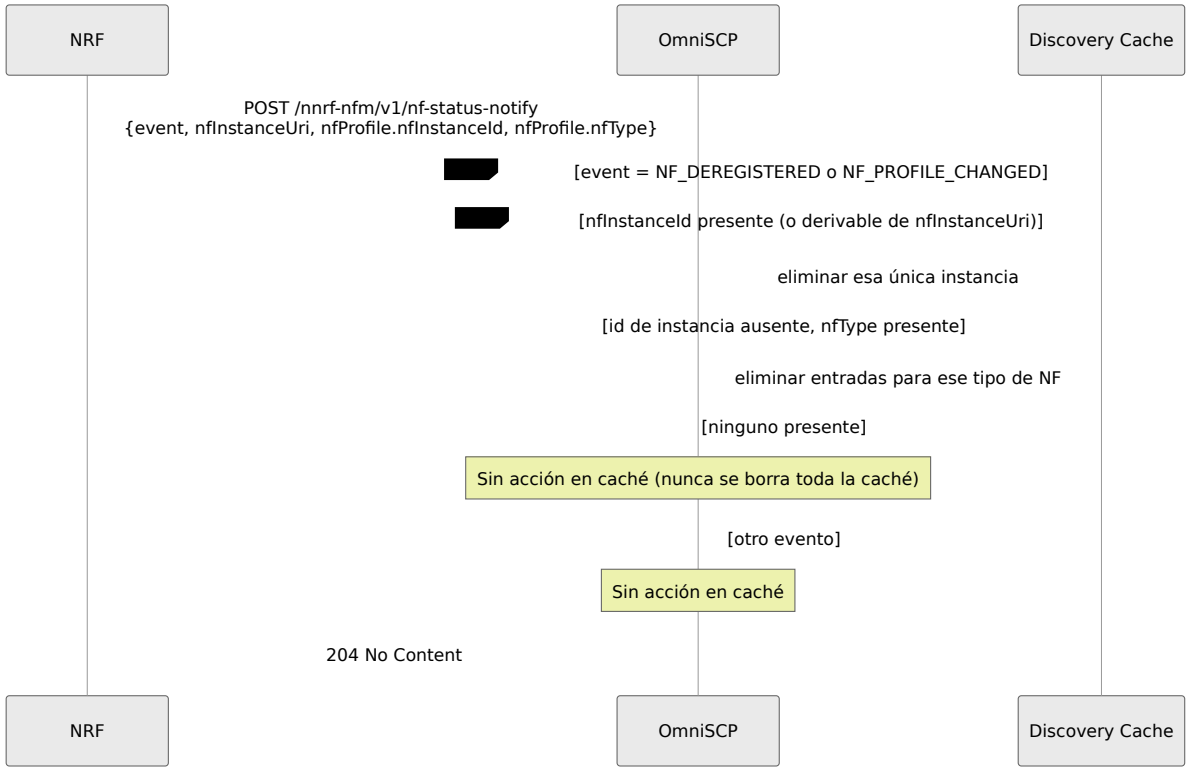


Reenvío directo (Modo 1)



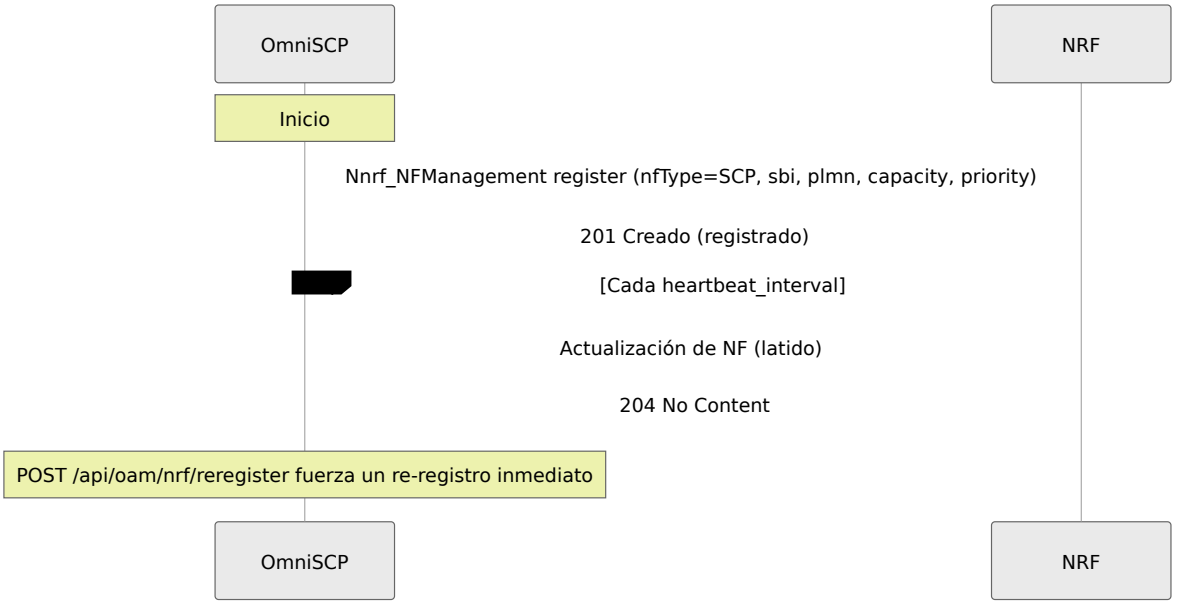
Sizing your network?
Get a deployment estimate in 60 seconds. Pick a workload size, see the topology, and get pricing — no sales call needed.
Sizing my network
Talk to engineering

Notificación de estado del NRF (invalidación de caché)



Auto-registro y latido del NRF

El SCP se registra con el NRF como un NF `SCP` al iniciar y mantiene el registro vivo con latidos periódicos. El perfil anunciado lleva la dirección/puerto/esquema SBI, el PLMN (`mcc`/`mnc`), y la `capacity`/`priority` de NF. El registro puede forzarse inmediatamente a través de la API de OAM.



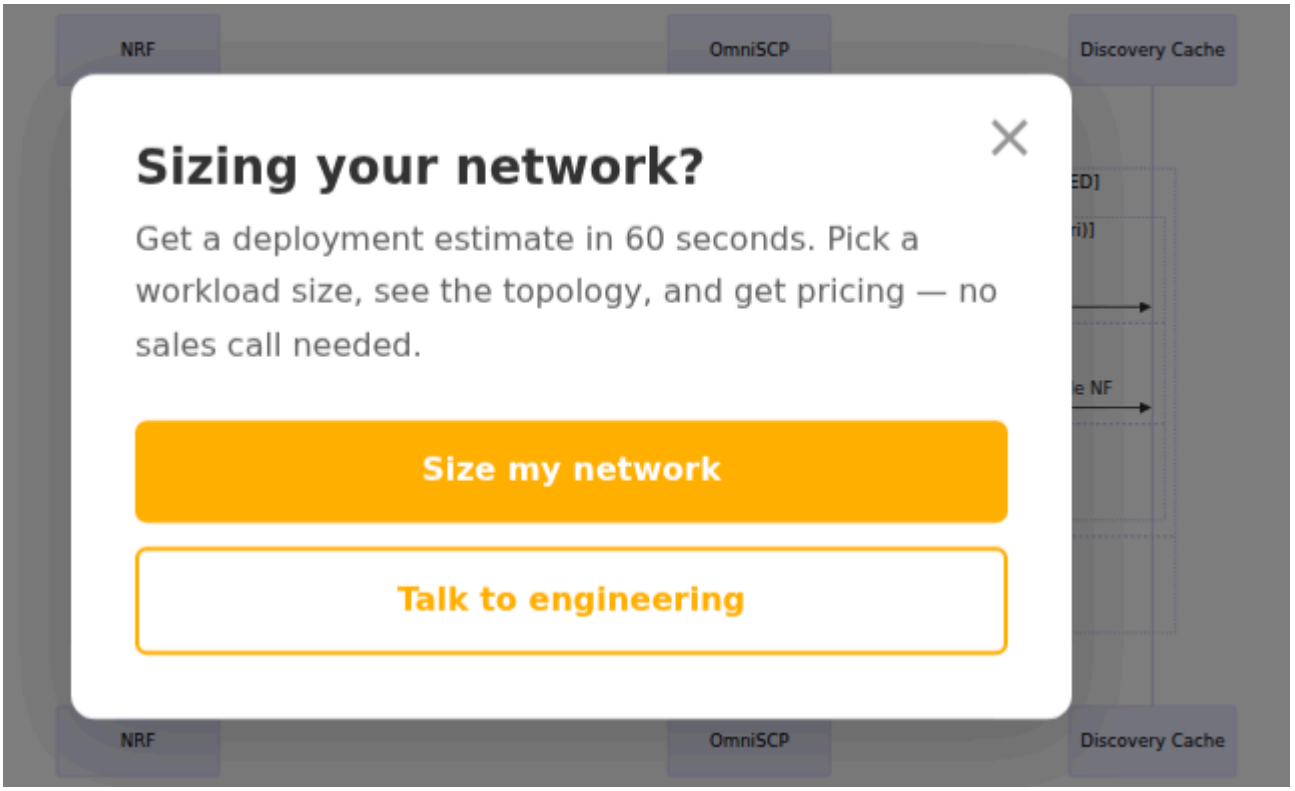
Drenaje administrativo upstream

El drenaje es un estado administrativo, separado del seguimiento de salud basado en fallos automático. Las instancias drenadas se filtran de la lista de candidatos *antes* de la selección del balanceador de carga, por lo que un productor drenado nunca recibe tráfico reenviado. Úselo para sacar un productor de la rotación de manera ordenada (para mantenimiento) sin desregistrarlo del NRF.

Flujo de trabajo de mantenimiento. Obtenga el `nfInstanceId` del productor objetivo del NRF (el valor con el que se registró el productor), luego `POST /api/oam/upstream/drain` con ese ID y un `reason` opcional. Las solicitudes en curso ya reenviadas al productor no se interrumpen; solo las *nuevas* selecciones lo omiten, por lo que permita que las solicitudes existentes se completen antes de atender al productor. Confirme que está fuera de rotación con `GET /api/oam/upstream/drain`, y cuando el mantenimiento haya terminado, `DELETE /api/oam/upstream/drain/{id}` para devolverlo al servicio.

> **El estado de drenaje está en memoria.** Las entradas de drenaje se mantienen en una tabla ETS y **no se persisten a través de un reinicio del SCP**: si el proceso SCP se reinicia mientras un productor está drenado, ese productor se vuelve seleccionable nuevamente y debe volver a emitir el drenaje. Planifique las ventanas de mantenimiento en consecuencia y

> vuelva a verificar `GET /api/oam/upstream/drain` después de cualquier reinicio del SCP.



← Resumen

Solución de problemas de OmniSCP

Las causas de error y los códigos de estado referenciados aquí se describen en Operaciones → Respuestas de error del proxy. Las claves de configuración y los puntos finales de OAM están en Configuración; métricas en Métricas.

400 Solicitud incorrecta: MANDATORY_IE_MISSING

Síntomas: El consumidor recibe `400` con causa `MANDATORY_IE_MISSING`.

Causas posibles:

- La solicitud no contenía ni `3gpp-Sbi-Target-apiRoot` ni el par `3gpp-Sbi-Discovery-target-nf-type` + `3gpp-Sbi-Discovery-service-names`.
- Dependiendo de la inferencia basada en la ruta (Modo 3) para un prefijo de servicio que no está en el mapa incorporado.

Resolución:

1. Confirme que el consumidor envía un target-apiRoot o ambos encabezados de descubrimiento.
2. Si depende de la inferencia de ruta, verifique que el prefijo de ruta esté en la tabla del Modo 3. Para un prefijo que no esté en esa tabla, envíe encabezados de descubrimiento explícitos.

504 Tiempo de espera de puerta de enlace: NF_DISCOVERY_FAILURE

Síntomas: El consumidor recibe `504` con causa `NF_DISCOVERY_FAILURE`.

Causas posibles:

- El NRF no devolvió instancias para el tipo de NF/servicio solicitado.
- El NRF no es accesible.

Resolución:

1. Verifique `nrf_uri` y la conectividad de red al NRF.
2. Confirme que el tipo de NF objetivo esté registrado en el NRF.
3. Verifique si una notificación de estado reciente del NRF invalidó la caché para el tipo de NF y el productor no se ha vuelto a registrar.

504 Tiempo de espera de puerta de enlace: NF_DISCOVERY_FAILURE (todas las instancias drenadas)

Síntomas: El consumidor recibe `504` con causa `NF_DISCOVERY_FAILURE` para un servicio que tiene productores registrados.

Causas posibles:

- Cada instancia candidata para el servicio está drenada administrativamente, por lo que el proxy no tiene un siguiente salto utilizable. Esta condición se mapea a `504 NF_DISCOVERY_FAILURE`, el mismo estado que el descubrimiento que no produce instancias.

Resolución:

1. Verifique `GET /api/oam/upstream/drain`. Si todas las instancias para el servicio están drenadas, desdrene las necesarias a través de `DELETE /api/oam/upstream/drain/{id}`.
2. Si no hay instancias drenadas y el servicio está registrado, trátelo como un caso de fallo de descubrimiento (vea la sección NF_DISCOVERY_FAILURE arriba).

500 Error interno del servidor: SYSTEM_FAILURE

Síntomas: El consumidor recibe `500` con causa `SYSTEM_FAILURE`.

Causas posibles:

- Un error interno del proxy genuinamente inesperado que no está cubierto por las causas más específicas (`400`, `502`, `504`) anteriores.

Resolución:

1. Revise los registros de SCP para obtener detalles sobre el error interno.

502 Puerta de enlace incorrecta: TARGET_NF_NOT_REACHABLE

Síntomas: El consumidor recibe `502` con causa `TARGET_NF_NOT_REACHABLE`.

Causas posibles:

- No se pudo resolver ninguna URI SBI a partir de un perfil de NF descubierto.
- Todos los productores seleccionados devolvieron errores de conexión después de reintentos.

Resolución:

1. Confirme que los productores estén en funcionamiento y sean accesibles en las direcciones SBI en sus perfiles NRF.
2. Si los productores son lentos, aumente `upstream_timeout`.
3. Revise `max_retries`; `0` convierte un solo fallo en un error inmediato.
4. Verifique `GET /api/api/health` para instancias marcadas como no saludables después de fallos repetidos.

Tiempos de espera en upstream / alta latencia del proxy

Síntomas: Elevado `omni_scp_proxy_request_duration_ms`; las solicitudes fallan intermitentemente con resultado `error`.

Causas posibles:

- Productores que responden más lento que `upstream_timeout`.
- Baja tasa de aciertos de caché que obliga a un viaje de ida y vuelta al NRF en muchas solicitudes.

Resolución:

1. Inspeccione el histograma de latencia y la tasa de error por tipo de NF.
2. Compare `omni_scp_cache_hits_total` vs `omni_scp_cache_misses_total`; una alta tasa de fallos significa descubrimiento frecuente del NRF, así que considere aumentar `discovery_cache_ttl`.
3. Ajuste `upstream_timeout` para que coincida con los tiempos de respuesta realistas de los productores. Recuerde que un intento que excede el tiempo de espera agrega el tiempo de espera completo a la latencia de extremo a extremo antes de un reintento.

Enrutamiento obsoleto / obsolescencia de la caché de descubrimiento

Síntomas: Las solicitudes se envían a productores que se han movido o reiniciado, causando errores hasta que las entradas caducan.

Causas posibles:

- Un productor cambió de dirección o se reinició sin desregistrarse del NRF, por lo que el SCP mantiene un perfil obsoleto hasta que expire el TTL.

Resolución:

1. Reduzca `discovery_cache_ttl` para acortar la ventana de obsolescencia.
2. Asegúrese de que los productores se desregistren del NRF al apagarse para que el SCP reciba una notificación `NF_DEREGISTERED` e invalide el tipo de NF afectado.
3. Vacíe explícitamente a través de `DELETE /api/oam/cache/{nf_type}` (o `DELETE /api/oam/cache/all`).

Registro de NRF no mantenido

Síntomas: `omni_scp_nrf_registration_status` lee `0`.

Causas posibles:

- `nrf_uri` incorrecto o NRF inaccesible.
- Desajuste de PLMN (`mcc`/`mnc`) contra la configuración del NRF.

Resolución:

1. Verifique `nrf_uri` y conectividad.
2. Confirme que `mcc`/`mnc` coincidan con la configuración PLMN del NRF.
3. Fuerce un re-registro con `POST /api/oam/nrf/reregister` y revise los registros de inicio.