



1. `3gpp-Sbi-Target-apiRoot`   `URI` `NRF`
2. `3gpp-Sbi-Discovery-target-nf-type` `3gpp-Sbi-Discovery-service-names` `SCP` `NRF`
3. `SCP` `NF` `/nudm-` → `UDM`

3GPP

SCP NF	3GPP TS 23.501 7.3
SCP	3GPP TS 29.500 6.10
3gpp-Sbi-Discovery-*	3GPP TS 29.500 6.10.3
3gpp-Sbi-Target-apiRoot	3GPP TS 29.500 6.10.3.2
3gpp-Sbi-Producer-Id	3GPP TS 29.500 6.10.3.3
SCP	3GPP TS 29.500 6.10.4
NF	3GPP TS 29.510 6.2
NRF NF	3GPP TS 29.510 6.3
SBI	3GPP TS 29.500

SBI

OmniSCP `NF`

Method	Path	Response Code	Description
POST	/nnrf-nfm/v1/nf-status-notify	204	NRF NF status notification. Response codes: NF_DEREGISTERED, NF_PROFILE_CHANGED. Status: 204 OK.
*	/*	-	Wildcard path for general operations.

ProblemDetails

SCP ProblemDetails (TS 29.500) details:

HTTP Code	ProblemDetail	Description
400	MANDATORY_IE_MISSING	Missing mandatory IE in request. Example: 3gpp-Sbi-Target-apiRoot.
502	TARGET_NF_NOT_REACHABLE	Target NF is not reachable. Example: 5xx error on SBI URI.
504	NF_DISCOVERY_FAILURE	NF discovery failure. Example: NRF cannot find NF.
500	SYSTEM_FAILURE	System failure on SCP side.

3gpp-Sbi

Field	Description
3gpp-Sbi-Target-apiRoot	API Root URL
3gpp-Sbi-Discovery-target-nf-type	Target NF type (e.g., UDM)
3gpp-Sbi-Discovery-service-names	Service names
3gpp-Sbi-Discovery-requester-nf-type	Requester NF type
3gpp-Sbi-Discovery-target-plmn-list	Target PLMN list
3gpp-Sbi-Discovery-requester-snssai-list	Requester S-NSSAI list
3gpp-Sbi-Discovery-nf-set-id	NF set ID
3gpp-Sbi-Discovery-target-nf-instance-id	Target NF instance ID
3gpp-Sbi-Discovery-requester-nf-instance-id	Requester NF instance ID

3gpp-Sbi

Field	Description
3gpp-Sbi-Producer-Id	Producer ID (e.g., TS 29.500 6.10.3.3)



config/runtime.exs

```
config :omniscp,  
  sbi_scheme: "http",  
  sbi_addr: "127.0.0.200",  
  sbi_port: 7777,  
  nrf_uri: "http://127.0.0.10:7777",  
  mcc: "999",  
  mnc: "70",  
  heartbeat_interval: 10_000,  
  discovery_cache_ttl: 60_000,  
  lb_strategy: :round_robin,  
  max_retries: 1,  
  upstream_timeout: 5_000
```

□□□

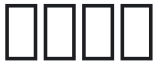
□□	□□□	□□	
sbi_scheme	"http"	□□ □	SBI □□□□□□□□□□
sbi_addr	"127.0.0.200"	□□ □	SBI HTTP □□□□□□□ □□□□□□□
sbi_port	7777	□□	SBI HTTP □□□□□□□
nrf_uri	"http://127.0.0.10:7777"	□□ □	NRF □□□□ URI□□□□ □□
mcc	"999"	□□ □	□□□□□□□□□□□□□□ □
mnc	"70"	□□ □	□□□□□□□□□□□□□□ □
heartbeat_interval	10_000	□□ □□ □□	NRF □□□□□□□□□□□
discovery_cache_ttl	60_000	□□ □□ □□	NRF □□□□□□□□□□□ service_name} [□ □□□□ 30 □□□□□□□□ □□□□□□□□□□□□□
lb_strategy	:round_robin	□□	NF □□□□□□□□□□□□ □□ :round_robin □□□□□□□□□□□□□□
max_retries	1	□□	□ NF □□□□□□ 5xx [□ □□□□□□□□□□□□□□□

項目	設定値	単位	説明
<code>upstream_timeout</code>	<code>5_000</code>	ms	NF へのリクエストのタイムアウト時間。HTTP リクエストの場合は、リクエストの送信と受信の両方のタイムアウト時間。

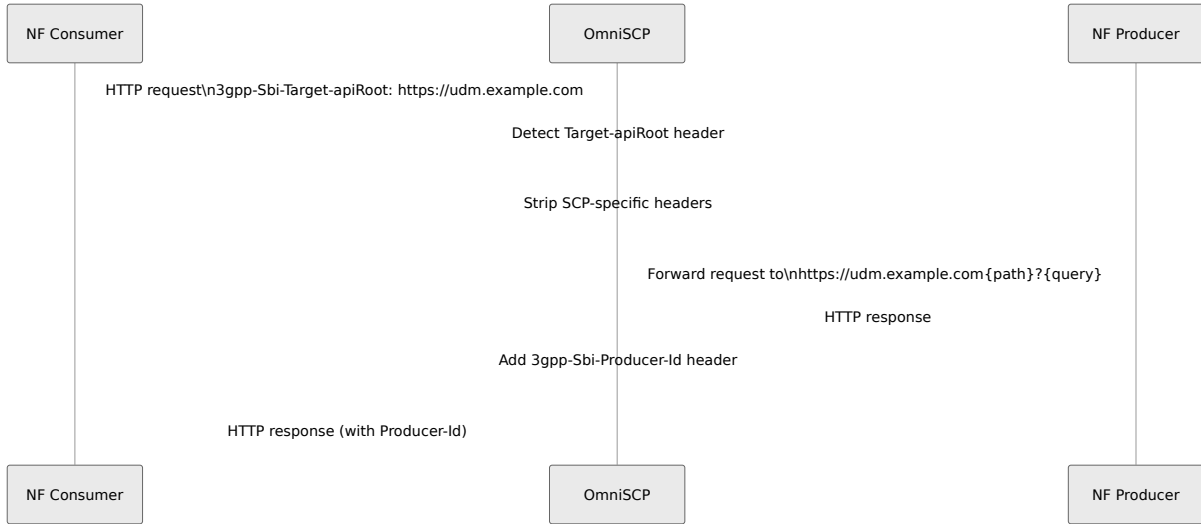
ロードバランシング

項目	説明
<code>:round_robin</code>	ラウンドロビン方式で、 <code>{nf_type, service_name}</code> のサービスに対して、登録された NF を順番にアクセスする。
<code>:weighted</code>	各 NF の <code>load - capacity</code> の値に基づいて、NRF から NF を選択する。 <code>load</code> と <code>capacity</code> の値は、サービス定義で指定する。
<code>:priority</code>	各 NF の <code>priority</code> の値に基づいて、サービス定義で指定する優先順位で NF を選択する。

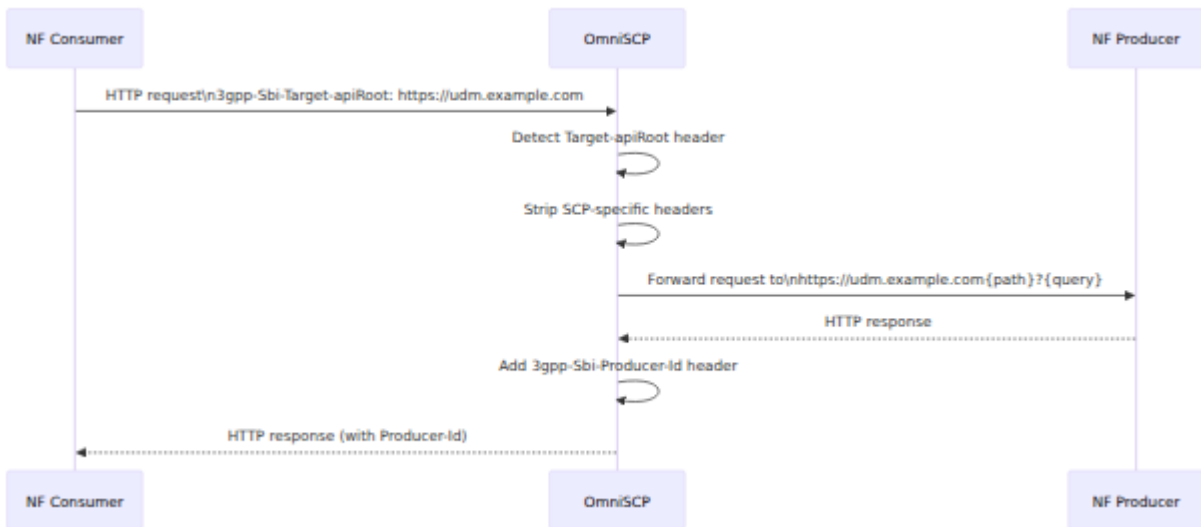
デフォルトはラウンドロビン方式で、サービスごとに 30 秒ごとにロードバランシングを実行する。



□□□□□□□□ 1□



□□□□□□□□□□ 2□



CHF- NEF- naf- SCP-L1 CHF NEF AF 3

CHF

CHF

CHF	CHF	CHF	CHF
[:omniscp, :proxy, :requests]	count, duration_ms	target_nf_type, result	CHF CHF
[:omniscp, :proxy, :result]	count, duration_ms	target_nf_type, result	CHF CHF
[:omniscp, :discovery, :cache]	hits, misses	target_nf_type, service_name	CHF CHF/CHF
[:omniscp, :cache, :hit]	count	—	CHF CHF
[:omniscp, :cache, :miss]	count	—	CHF CHF
[:omniscp, :associations, :active]	count	—	CHF CHF
[:omni5g, :nrf, :registration]	status	nf_type	NRF CHF 1=CHF 0=CHF

CHF success 2xx/3xx client_error 4xx server_error 5xx error CHF/CHF

Prometheus ☐☐

SCP ☐☐☐☐

☐☐	☐☐	☐☐
<code>omni_scp.proxy.requests.count</code>	counter	<code>target_nf_type,</code> <code>result</code>
<code>omni_scp.proxy.requests.duration_ms</code>	summary	<code>target_nf_type</code>
<code>omni_scp.proxy_requests.total</code>	counter	<code>target_nf_type,</code> <code>result</code>
<code>omni_scp.proxy_request.duration_ms</code>	distribution	<code>target_nf_type</code>
<code>omni_scp.active_associations.count</code>	gauge	--

☐☐☐☐

メトリック名	タイプ	ラベル	単位
<code>omni_scp.discovery.cache.hits</code>	counter	<code>target_nf_type,</code> <code>service_name</code>	個 個 個 個 個
<code>omni_scp.discovery.cache.misses</code>	counter	<code>target_nf_type,</code> <code>service_name</code>	個 個 個 個 個
<code>omni_scp.cache_hits.total</code>	counter	--	個 個 個 個 個
<code>omni_scp.cache_misses.total</code>	counter	--	個 個 個 個 個

NRF

メトリック名	タイプ	ラベル	説明
<code>omni_scp.nrf.registration.status</code>	gauge	<code>nf_type</code>	NRF 登録状態 1=登録済み 0=未登録

BEAM VM

名前	タイプ	説明
<code>beam.memory.total</code>	gauge	BEAM の総メモリ
<code>beam.memory.processes</code>	gauge	Erlang のプロセス数
<code>beam.memory.processes_used</code>	gauge	プロセスに使用されているメモリ
<code>beam.memory.system</code>	gauge	システムメモリ
<code>beam.memory.atom</code>	gauge	アトムメモリ
<code>beam.memory.atom_used</code>	gauge	アトムに使用されているメモリ
<code>beam.memory.binary</code>	gauge	バイナリメモリ
<code>beam.memory.code</code>	gauge	コードメモリ
<code>beam.memory.ets</code>	gauge	ETS のメモリ
<code>beam.processes.count</code>	gauge	Erlang のプロセス数
<code>beam.ports.count</code>	gauge	Erlang のポート数
<code>beam.atom.count</code>	gauge	アトムの数
<code>beam.vm.uptime</code>	gauge	VM の稼働時間

□□□□

□□	□□	□□
info	Received NRF status notification	□□ NF □□□□
info	NRF notification: event=<E> nf=<URI>	□□□□□□□□
debug	SCP direct forward: <METHOD> <URL>	□□ 1 □□
debug	SCP delegated forward: <METHOD> <URL> (attempt <N>)	□□ 2/3 □□□ □
warning	SCP retrying after <STATUS> from <ID>...	□□ 5xx □□□ □□
warning	SCP retrying after error from <ID>...	□□□□□□□□□□ □□
warning	SCP cannot determine target for <METHOD> <PATH>	□□ 3 □□□□□□ □□□□
warning	NRF discovery returned no instances for <NF>/<SVC>	□□□□□□□□
warning	All NF instances unhealthy, falling back to full list	LB □□□□
error	NRF discovery failed: ...	NRF □□□□
error	SCP proxy error: ...	□□□□□□□□
info	NF instance <ID> recovered after cooldown	□□□□□□□□

2. 3 `nchf-` `nnef-` `naf-` SCP-L1

504 — NF_DISCOVERY_FAILURE

NRF NF

1. NRF OmniSCP `nrf_uri`
2. NF NRF `GET {nrf_uri}/nnrf-disc/v1/nf-instances?target-nf-type=<TYPE>`
3. NRF `NF_DEREGISTERED` NF

502 — TARGET_NF_NOT_REACHABLE

NF

1. NF NRF SBI
2. `upstream_timeout` NF
3. `max_retries` 0 502
4. `NF instance <ID> marked unhealthy after N failures`

NF NRF SBI URI TTL

1. `discovery_cache_ttl`
2. NF NRF NRF OmniSCP
3. OmniSCP

1. `omni_scp.proxy_request.duration_ms`
2. `omni_scp.cache_hits.total` `omni_scp.cache_misses.total` `discovery_cache_ttl`
3. `upstream_timeout` —

NRF 注册

omni_scp.nrf.registration.status 注册状态 0

1. nrf_uri 注册 NRF 注册
2. mcc 或 mnc 注册 NRF PLMN 注册
3. 注册 NRF 注册