OmniNEF Configuration Reference

All configuration lives under the `:omninef` application key. Compile-time defaults are in `config/config.exs` (mirrored by `OmniNef.Config`); production deployments override a subset via environment variables in `config/runtime.exs`.

Table of Contents

SBI Listener

Key	Type	Default	Description
`sbi_scheme`	string	`"http"`	Scheme advertised in the NRF profile. The listener itself serves plain HTTP.
`sbi_addr`	string	`"127.0.0.30"`	SBI bind/advertise address.
`sbi_port`	integer	`7777`	SBI listen port (northbound API + status).

NRF and PLMN

Key	Type	Default	
`nrf_uri`	string	`"http://127.0.0.10:7777"`	Base URI of t discovery.
`mcc`	string	`"999"`	Served PLMN profile and th
`mnc`	string	`"70"`	Served PLMN
`heartbeat_interval`	integer (ms)	`10_000`	NRF heartbea
`nef_fqdn`	string	derived	This NEF's FC `nef.5gc.mnc`

AF Registry

`af_registry` is a map keyed by the AF's API key. Each value scopes what the AF may do. It is empty by default: **no AF can call the NEF until one is provisioned**.

```
config :omninef,
  af_registry: %{
    "af-secret-001" => %{
      af_id: "af-acme",              # identifier echoed into logs
      allowed_apis: :all,            # or a list of API atoms (see
below)
      rate_limit_per_sec: 20         # optional; falls back to
default_rate_limit_per_sec
    }
  }
```

`allowed_apis` is either `:all` or a list drawn from:

API atom	Northbound resource
`:monitoring_event`	`/3gpp-monitoring-event/...`
`:parameter_provision`	`/nnef-pp/...`
`:af_session_with_qos`	`/3gpp-as-session-with-qos/...`
`:traffic_influence`	`/3gpp-traffic-influence/...`

The API key is presented by the AF as `Authorization: Bearer <key>` or `X-Api-Key: <key>`.

Throttling

Key	Type	Default	Description
`default_rate_limit_per_sec`	integer	`10`	Per-AF request ceiling applied when an AF spec omits `rate_limit_per_sec`.

Throttling uses a sliding one-second window per AF. An AF at or over its ceiling receives `429`.

Environment Variables

Set in `config/runtime.exs` for `MIX_ENV=prod`. Unset variables fall back to the compile-time defaults above.

Variable	Maps to	
`NEF_SBI_SCHEME`	`sbi_scheme`	`http`
`NEF_SBI_ADDR`	`sbi_addr`	`127.0.`
`NEF_SBI_PORT`	`sbi_port`	`7777`
`NEF_NRF_URI`	`nrf_uri`	`http:/`
`NEF_MCC`	`mcc`	`999`
`NEF_MNC`	`mnc`	`70`
`NEF_FQDN`	`nef_fqdn`	derived
`NEF_DEFAULT_RATE_LIMIT_PER_SEC`	`default_rate_limit_per_sec`	`10`
`NEF_AF_REGISTRY`	`af_registry` (JSON)	unset (

`NEF_AF_REGISTRY` is a JSON object of the same shape as the `af_registry` map, with `allowed_apis` as the string `"all"` or an array of API-name strings:

```
NEF_AF_REGISTRY='{"af-secret-001":{"af_id":"af-
acme","allowed_apis":"all","rate_limit_per_sec":20}}'
```

Malformed JSON raises at boot so a bad deployment config fails fast rather than silently disabling all AFs.

OmniNEF Metrics and Monitoring

Current Status

Observability is provided by **structured application logs** and standard request logging. Operators monitor OmniNEF through its logs, which capture authorization, throttling, and translation activity across the northbound and southbound interfaces.

What to Watch in the Logs

The application logs are the primary signal for operating OmniNEF.

Authorization and Throttling

Rejected northbound requests surface as `ProblemDetails` responses. Watch for:

- A rising rate of `401 UNAUTHORIZED`: an AF presents a missing or unknown API key (a misconfigured client or an unprovisioned AF).
- `403 OPERATION_NOT_ALLOWED`: a known AF calls an API it is not entitled to. Check the AF's `allowed_apis`.
- `429 TOO_MANY_REQUESTS`: an AF exceeds its request ceiling. Consider raising its `rate_limit_per_sec` or investigate the client.

Northbound-to-Southbound Translation

Each capability logs the operation it performs, including the target UE identifier and the resolved producer NF. Watch these to confirm subscriptions and sessions are reaching the UDM/PCF/UDR.

- `503 NF_UNAVAILABLE` indicates NRF discovery found no suitable UDM/PCF/UDR instance.
- `502 TARGET_NF_NOT_REACHABLE` indicates the producer NF was discovered but the SBI call failed.
- `400 MANDATORY_IE_INCORRECT` indicates a request-validation failure (missing UE identifier, QoS, routing target, or an unsupported monitoring type).

Request Logging

Standard request logging covers each northbound SBI request handled by the listener. Use it to observe request volume, methods and paths, and to correlate AF activity with translation events.

Health Checking

Health is best assessed by combining:

- **Liveness:** `GET /nnef/v1/status` returns `200 {"status":"UP"}` when the SBI listener is serving. See the Operations Guide.
- **NRF registration:** confirm OmniNEF appears in the NRF as a `NEF` NF instance and is heartbeating. Registration behaviour is described in the Operations Guide.
- **Log signals:** the authorization, throttling and translation events described above.

OmniNEF Operations Guide

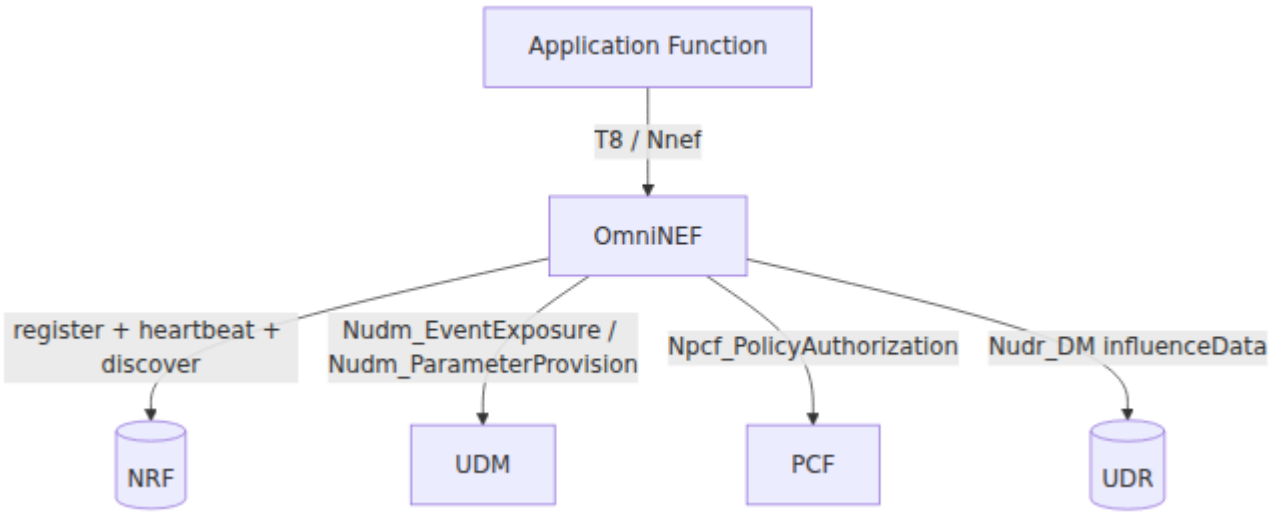
Overview | Configuration Reference | Metrics and Monitoring | Troubleshooting

This guide describes how OmniNEF behaves as the 5G Core's northbound exposure gateway and how to operate it. OmniNEF is the Network Exposure Function defined in 3GPP TS 23.501 §6.2.5, exposing the T8 API (TS 29.122) / Nnef services (TS 29.522) toward Application Functions.

Table of Contents

Architecture Overview

OmniNEF terminates the northbound API toward external AFs and translates each request into an SBI operation toward a core producer NF. It registers with the NRF as NFType `NEF` and uses NRF discovery to resolve the UDM, PCF or UDR that serves a given request.



Request Lifecycle

Every northbound resource-creation request follows the same path:

1. **Authorize** the calling AF from its API key and check it is entitled to the requested API.
2. **Throttle** it against the AF's per-second request ceiling.
3. **Validate** the request body (target UE identifier, required QoS, routing target, …).
4. **Discover** the target NF (UDM/PCF/UDR) from the NRF and select a load-aware instance.
5. **Translate and send** the corresponding SBI request to the producer NF.
6. **Store** the created resource (owning AF, representation, downstream tear-down URL) and return `201 Created` with a `Location` header pointing at the individual resource.

`GET` on the individual resource returns the stored representation; `DELETE` tears the resource down in the producer NF (best-effort) and removes the NEF-side record.

Authorization and Throttling

The NEF is the boundary between untrusted third-party AFs and the trusted core, so authorization is mandatory (TS 23.501 §6.2.5).

- **Authentication:** the AF presents an API key as `Authorization: Bearer <key>` or `X-Api-Key: <key>`. Keys are looked up in the AF registry. A missing or unknown key returns `401`.
- **Entitlement:** each AF is scoped to `:all` APIs or an explicit list. A known AF calling an API it is not entitled to returns `403`.
- **Throttling:** each AF has a per-second request ceiling (its own or the configured default). An AF exceeding it returns `429`. The window is a sliding one second.

See the Configuration Reference for how to onboard an AF.

Northbound APIs

Capability	Northbound resource	Method
Monitoring Events (Nnef_EventExposure)	`/3gpp-monitoring-event/v1/{scsAsId}/subscriptions`	POST, G DELETE
Parameter Provisioning (Nnef_ParameterProvision)	`/nnef-pp/v1/{afId}/provisionings`	POST, G DELETE
AF session with QoS (Nnef_AFsessionWithQoS)	`/3gpp-as-session-with-qos/v1/{scsAsId}/subscriptions`	POST, G DELETE
Traffic Influence (Nnef_TrafficInfluence)	`/3gpp-traffic-influence/v1/{afId}/subscriptions`	POST, G DELETE

Monitoring Events

An AF subscribes to network events for a UE identified by `externalId` or `msisdn`. The supported `monitoringType` values map to UDM event types:

T8 `monitoringType`	UDM `eventType`
`LOSS_OF_CONNECTIVITY`	`LOSS_OF_CONNECTIVITY`
`UE_REACHABILITY`	`UE_REACHABILITY_FOR_DATA`
`LOCATION_REPORTING`	`LOCATION_REPORTING`
`ROAMING_STATUS`	`ROAMING_STATUS`
`COMMUNICATION_FAILURE`	`COMMUNICATION_FAILURE`
`AVAILABILITY_AFTER_DDN_FAILURE`	`AVAILABILITY_AFTER_DDN_FAILURE`

The external identifier is mapped to a UDM GPSI (`extid-<externalId>` or `msisdn-<msisdn>`).

Parameter Provisioning

An AF provisions expected-UE-behaviour or communication-characteristics parameters for a UE (`externalId` / `msisdn`) or a group (`externalGroupId`). Either an explicit `ppData` object or the recognised top-level provisioning fields are accepted; the NEF applies them via `Nudm_ParameterProvision`, which the UDM persists to the UDR.

AF session with QoS

An AF supplies the UE address (`ueIpv4Addr` / `ueIpv6Addr`), the IP flows to treat, and a `qosReference` (or requested QoS). The NEF creates an Individual Application Session Context at the PCF, which installs the corresponding policy toward the SMF.

Traffic Influence

An AF asks the core to steer selected traffic toward a DNAI / application location. The routing target depends on scope:

- **Individual UE** (`gpsi` or `supi` present) → an application session at the **PCF**.
- **Group / any-UE** (no individual identifier) → stored as `influenceData` application data in the **UDR**, which the PCF reads when building policy for future matching sessions.

Southbound Interworking

The NEF resolves each producer NF from the NRF (`Nnrf_NFDiscovery`, service-name filtered) and selects the lowest-load instance. It then issues the mapped SBI request:

- `POST …/nudm-ee/v1/{gpsi}/ee-subscriptions` (UDM)
- `PUT …/nudm-pp/v1/{gpsi}/pp-data` (UDM)
- `POST …/npcf-policyauthorization/v1/app-sessions` (PCF)
- `PUT …/nudr-dr/v1/application-data/influenceData/{id}` (UDR)

When a producer NF returns a `Location`, the NEF retains the resolved absolute URL so the resource can be deleted later. For monitoring events the NEF registers a **per-subscription** notification callback URL with the producer NF (`…/nnef-eventexposure/v1/notifications/{subId}`); when the producer POSTs an event report to that callback, the NEF maps it to a T8 `MonitoringNotification` and relays it to the AF's stored `notificationDestination` (the event-exposure data path). Relay is best-effort. The NEF always acknowledges the producer callback with `204`. The NEF drops notifications for an unknown or expired subscription. See the notification-relay troubleshooting entry.

SBI Endpoints

- `GET /nnef/v1/status`: liveness probe; returns `200 {"status":"UP"}`.
- The northbound resource endpoints in the Northbound APIs table.

Unmatched paths return `404` with a 3GPP `ProblemDetails` body.

NRF Registration

At start-up OmniNEF registers with the NRF as NFType `NEF`, advertising four services (`nnef-eventexposure`, `nnef-pp`, `nnef-afsessionwithqos`, `nnef-trafficinfluence`) and its SBI IP endpoint, then heartbeats on the configured interval. Capacity and priority come from the shared `Omni5gEx.NRF.Config` helpers.

Operational Notes

- **Plain HTTP:** the SBI listener serves plain HTTP. Terminate TLS externally (a reverse proxy or service mesh) for any exposure beyond a trusted network.
- **Non-persistent state:** the AF registry, rate-limiter windows and created-resource store are in-memory. On restart, AFs are re-seeded from configuration and previously created northbound resources are forgotten by the NEF (the underlying core-NF resources persist until they expire or are removed directly).
- **Downstream tear-down is best-effort:** a `DELETE` always removes the NEF-side record even if the producer NF is briefly unreachable; the failure is logged.

OmniNEF Troubleshooting

Table of Contents

Common Issues

Every AF request returns 401

The AF's API key is not in the registry. The registry is empty by default, so a fresh deployment rejects everything until an AF is provisioned. Add the AF to `af_registry` (or `NEF_AF_REGISTRY`) and confirm the client sends `Authorization: Bearer <key>` or `X-Api-Key: <key>`. See the Configuration Reference.

An AF gets 403 on some APIs

The AF is known but its `allowed_apis` does not include the API it is calling. Set `allowed_apis` to `:all`, or add the relevant API atom to the list.

An AF gets 429 under load

The AF is exceeding its per-second request ceiling. Raise its `rate_limit_per_sec` (or `default_rate_limit_per_sec`), or throttle the client. The window is a sliding one second.

Requests fail with 503 NF_UNAVAILABLE

NRF discovery returned no UDM/PCF/UDR instance offering the required service. Confirm the target NF is registered with the NRF and healthy, and that `nrf_uri` is correct.

Requests fail with 502 TARGET_NF_NOT_REACHABLE

The producer NF was discovered but the SBI call to it failed. Check connectivity to the producer NF and its logs; the NEF logs the underlying error.

An AF creates a monitoring-event subscription but never receives notifications

Event notifications now follow the data path end to end: on create, the NEF registers a **per-subscription** callback with the UDM (`.../nnef-eventexposure/v1/notifications/{subId}`); when the producer NF POSTs an event report to that callback, the NEF maps it to a T8 `MonitoringNotification` and relays it to the AF's `notificationDestination`. If the AF sees nothing, check, in order:

- The subscription carried a reachable `notificationDestination` (the NEF logs `subscription <id> has no notificationDestination` when it is absent).
- The producer NF is actually emitting reports and can reach the NEF's callback URL (the NEF's advertised `sbi_base_url` must be routable from the UDM/AMF).
- The relay POST succeeds. The NEF logs a failed delivery as `relay to AF <url> ... failed`. Relay is best-effort: the NEF always answers the producer callback with `204` so the producer does not treat a transiently-down AF as a permanent callback failure. Notifications for an unknown/expired subscription are logged and dropped (also `204`).

Requests fail with 400

The request body is missing a mandatory field for that capability: a UE identifier (`externalId`/`msisdn`), a `qosReference` for an AF QoS session, a routing target for traffic influence, or an unsupported `monitoringType`. The `detail` field of the `ProblemDetails` names the specific validation failure.