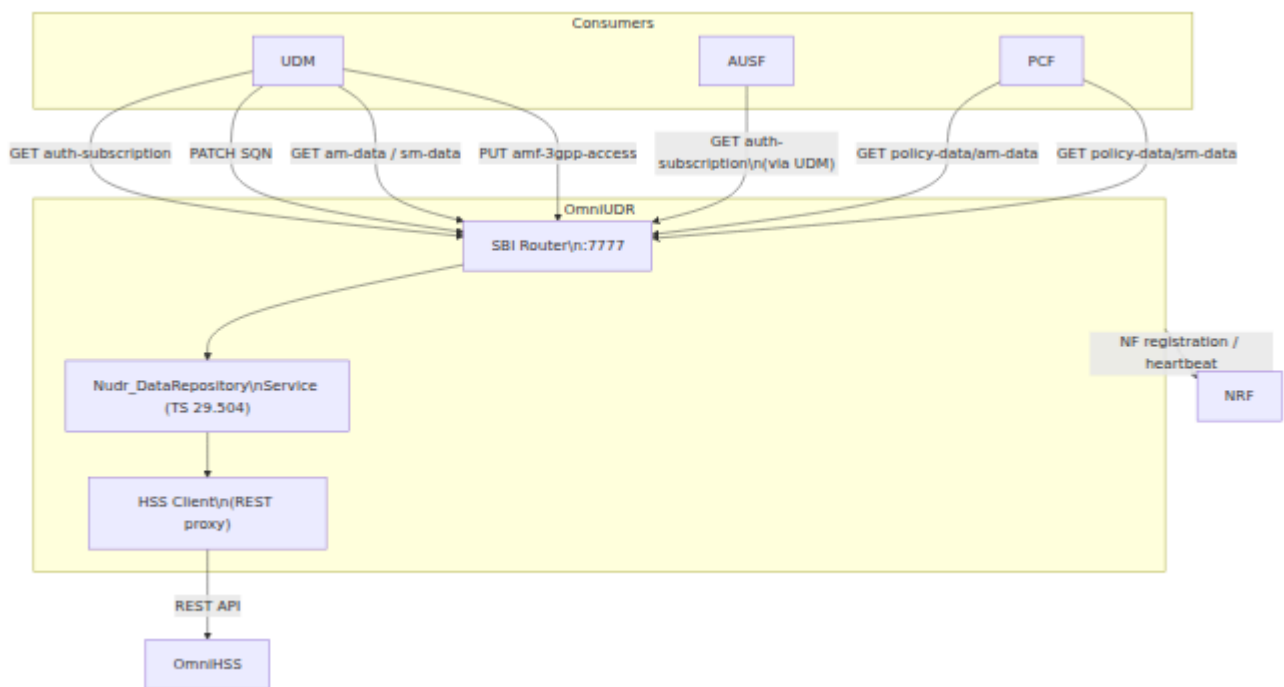


OmniUDR Operations

1. Component Overview

OmniUDR is the standalone Unified Data Repository (UDR) for the Omnitouch 5G core. It implements the Nudr_DataRepository service (TS 29.504), exposing subscriber data and policy data to UDM, AUSF, and PCF consumers via SBI. OmniUDR proxies all data requests to OmniHSS via the HSS REST API, acting as a 3GPP-compliant facade over the legacy subscriber store.

OmniUDR was previously an embedded proxy layer within OmniUDM and is now deployed as an independent NF with its own SBI endpoint and NRF registration.



2. 3GPP Role and Spec References

Aspect	Reference
UDR functional definition	TS 23.501 Section 6.2.9
Nudr_DataRepository service	TS 29.504
Subscription data model	TS 29.505
Policy data model	TS 29.519
Authentication subscription data	TS 29.504 Section 5.2.2
Provisioned data (AM, SM)	TS 29.504 Section 5.2.3
Context data (AMF/SMF registrations)	TS 29.504 Section 5.2.4
Policy data (AM/SM policy)	TS 29.504 Section 5.2.5

3. SBI Endpoints

All endpoints are HTTP/1.1 with `Content-Type: application/json`.

Subscription Data (TS 29.504)

Method	Path	Description	
GET	/nudr-dr/v2/subscription-data/{ueId}/authentication-data/authentication-subscription	Retrieve authentication subscription (Ki, OPc, SQN)	
PATCH	/nudr-dr/v2/subscription-data/{ueId}/authentication-data/authentication-subscription	Update authentication subscription (SQN update)	
GET	/nudr-dr/v2/subscription-data/{ueId}/{servingPlmnId}/provisioned-data/am-data	Retrieve provisioned AM data	
GET	/nudr-dr/v2/subscription-data/{ueId}/{servingPlmnId}/provisioned-data/sm-data	Retrieve provisioned SM data	
GET	/nudr-dr/v2/subscription-data/{ueId}/{servingPlmnId}/provisioned-data/smf-select-data	Retrieve SMF selection data	

Context Data (TS 29.504)

Method	Path	Description	Success
PUT	/nudr-dr/v2/subscription-data/{ueId}/context-data/amf-3gpp-access	Store AMF registration context	204 No Content
PUT	/nudr-dr/v2/subscription-data/{ueId}/context-data/smf-registrations/{pduSessionId}	Store SMF registration context	204 No Content

Policy Data (TS 29.519)

Method	Path	Description	Success
GET	/nudr-dr/v2/policy-data/ues/{ueId}/am-data	Retrieve AM policy data	200 OK
GET	/nudr-dr/v2/policy-data/ues/{ueId}/sm-data	Retrieve SM policy data	200 OK

4. Configuration Reference

OmniUDR is configured via Elixir application environment under the `:omniudr` key.

Example Configuration

```
config :omniudr,  
  sbi_scheme: "http",  
  sbi_addr: "127.0.0.22",  
  sbi_port: 7777,  
  nrf_uri: "http://127.0.0.10:7777",  
  mcc: "999",  
  mnc: "70",  
  heartbeat_interval: 10_000,  
  hss_api_base_url: "https://127.0.0.1:8443"
```

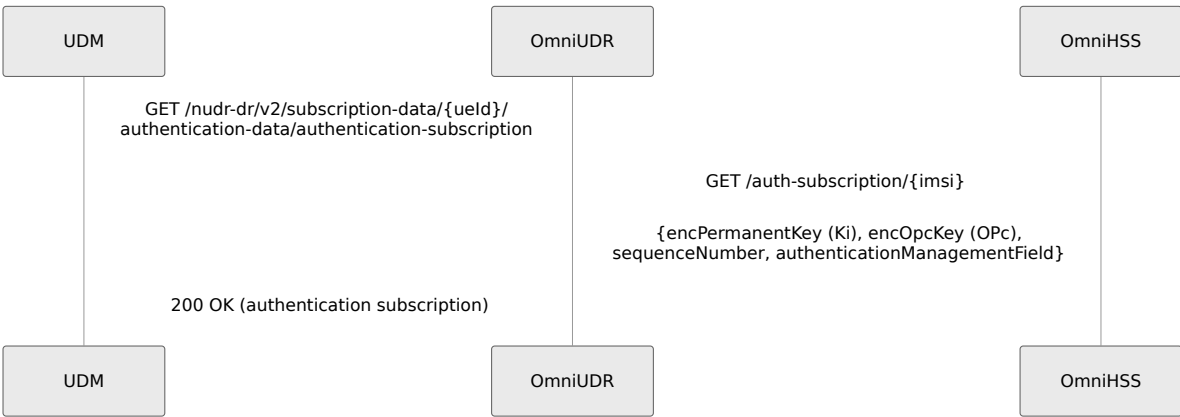
Parameter Table

Parameter	Type	Default	Description
<code>sbi_scheme</code>	string	<code>"http"</code>	URI scheme for the SBI HTTP server
<code>sbi_addr</code>	string	<code>"127.0.0.22"</code>	IP address the SBI HTTP server binds to
<code>sbi_port</code>	integer	<code>7777</code>	TCP port the SBI HTTP server listens on
<code>nrf_uri</code>	string	<code>"http://127.0.0.10:7777"</code>	Base URI of the NRF for NF registration and heartbeat
<code>mcc</code>	string	<code>"999"</code>	Mobile Country Code for the serving PLMN
<code>mnc</code>	string	<code>"70"</code>	Mobile Network Code for the serving PLMN

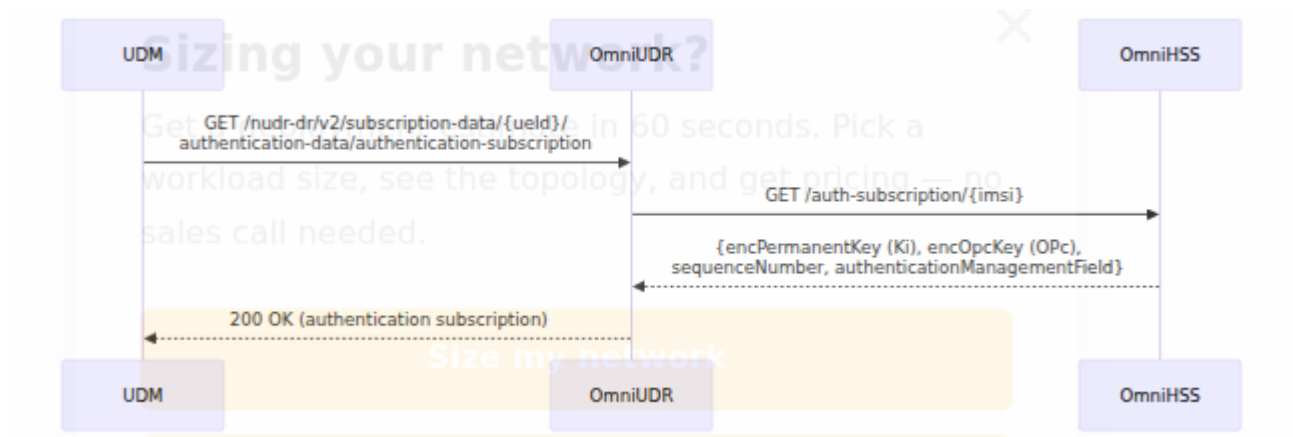
Parameter	Type	Default	Description
<code>heartbeat_interval</code>	integer (ms)	<code>10000</code>	Interval at which OmniUDR sends NRF heartbeat PATCH requests
<code>hss_api_base_url</code>	string	<code>"https://127.0.0.1:8443"</code>	Base URL of the OmniHSS REST API. All data requests are proxied to this backend

5. Key Procedures

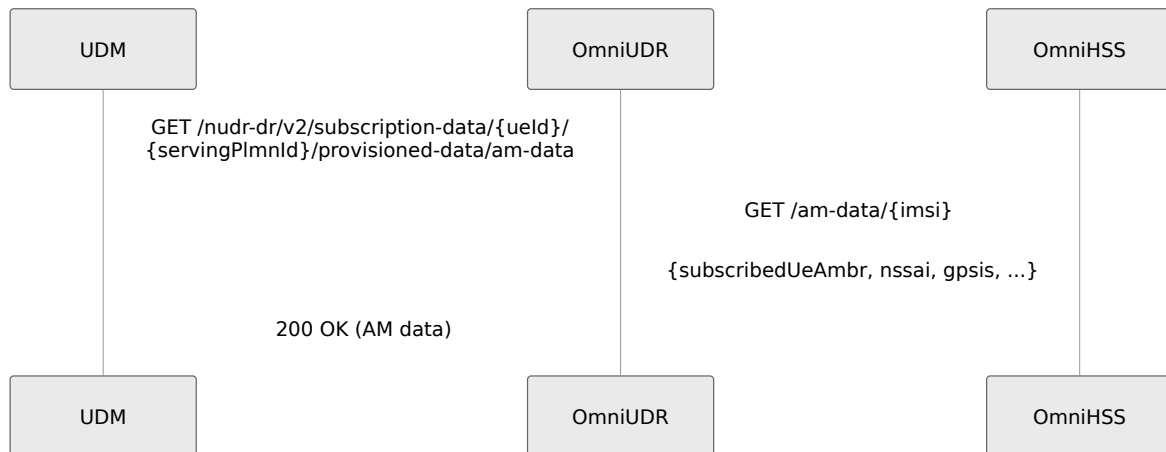
5.1 Authentication Subscription Retrieval



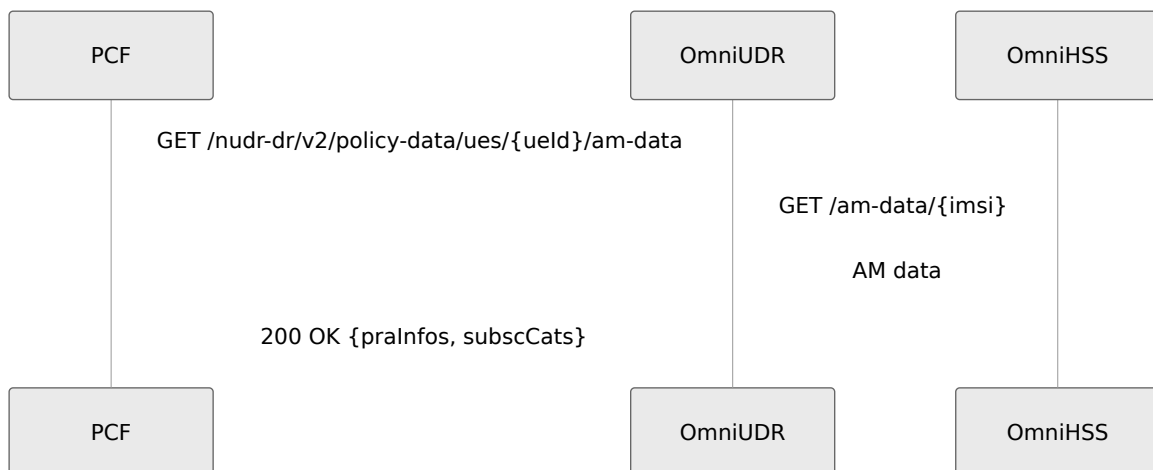
5.2 SQN Update



5.3 Provisioned Data Retrieval



5.4 Policy Data Retrieval



6. Prometheus Metrics

UDR Metrics

Metric	Type	Tags	Description
<code>omni_udr.nrf.registration.status</code>	gauge	<code>nf_type</code>	NRF registration status (1=registered, 0=not)
<code>omni_udr.hss.health</code>	gauge	--	HSS connectivity health (1=up, 0=down)
<code>omni_udr.hss_requests.total</code>	counter	<code>endpoint</code> , <code>result</code>	Total requests to HSS backend
<code>omni_udr.hss_request.duration_ms</code>	distribution	<code>endpoint</code>	HSS request duration in milliseconds (buckets: 5, 10, 25, 50, 100, 250, 500, 1000, 2500)

BEAM VM Metrics

Metric	Type	Description
beam.memory.total	gauge	Total BEAM memory in bytes
beam.memory.processes	gauge	Memory used by Erlang processes
beam.memory.system	gauge	System memory (ETS, atoms, code)
beam.processes.count	gauge	Number of Erlang processes
beam.vm.uptime	gauge	VM uptime in seconds

7. Known Limitations

ID	Area	Description
UDR-1	Data model	The UDR data model proxied from OmniHSS is partially implemented. Authentication subscription, AM data, and SM data are functional. Some provisioned data fields may be absent from HSS responses
UDR-2	Policy data	Policy data endpoints (<code>/policy-data/ues/{ueId}/am-data</code> and <code>.../sm-data</code>) return minimal structures derived from HSS AM/SM data. Full TS 29.519 policy data model is not supported
UDR-3	Subscriptions	Data change subscription and notification (<code>Nudr_DataRepository_Subscribe</code>) is not implemented. Consumers cannot subscribe to data change events
UDR-4	SMF registration	<code>put_smf_registration</code> stores context locally but does not forward to OmniHSS
UDR-5	In-memory proxy	OmniUDR is a stateless proxy; all data at rest is owned by OmniHSS. No local caching or persistence

8. Troubleshooting

GET returns 404 for a known subscriber

OmniUDR proxies the request to OmniHSS. Confirm:

1. `hss_api_base_url` is reachable from the OmniUDR host.

2. The subscriber IMSI exists in OmniHSS.
3. The `ueId` in the request path uses the `imsi-{digits}` format.

PATCH for SQN update returns 500

The HSS failed to accept the SQN update. Check OmniHSS logs. The UDR logs `Failed to update SQN for {ueId}: {reason}` but returns `:ok` to avoid blocking authentication flows.

Policy data queries return minimal data

The `/policy-data/` endpoints derive policy structures from HSS AM/SM data rather than a dedicated policy store. This is by design in the current architecture. Full per-subscriber policy data requires a dedicated policy data store (not yet implemented).

HSS health check

Monitor `omni_uds.hss.health` gauge (1=up, 0=down) and `omni_uds.hss_request.duration_ms` for latency trends. High latency may indicate HSS overload or network issues.