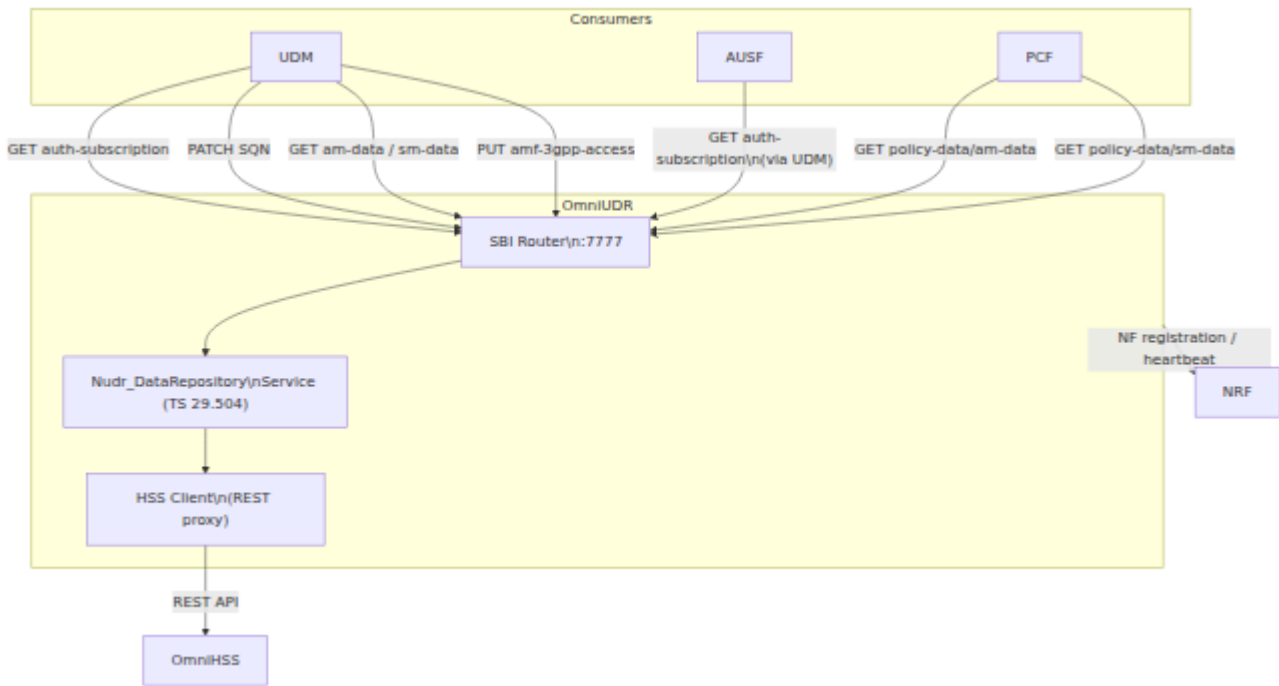


OmniUDR

1. Overview

OmniUDR is an Omnitech 5G Network Data Repository (UDR) implementation based on Nudr_DataRepository (TS 29.504) that serves as a central data repository for SBI, UDM, AUSF, and PCF. It provides a unified interface for these services to access and manage subscriber data. The implementation is based on the HSS REST API and OmniHSS, which is a 3GPP-compliant implementation.

OmniUDR is implemented as an SBI (Service Based Interface) and provides a unified interface for SBI consumers (UDM, AUSF, PCF) to access and manage subscriber data. It also provides a unified interface for SBI producers (Nudr_DataRepository, HSS Client) to store and retrieve subscriber data.



2. 3GPP

Item	Item
UDR	TS 23.501 6.2.9
Nudr_DataRepository	TS 29.504
	TS 29.505
	TS 29.519
	TS 29.504 5.2.2
(AM, SM)	TS 29.504 5.2.3
(AMF/SMF)	TS 29.504 5.2.4
(AM/SM)	TS 29.504 5.2.5

3. SBI

HTTP/1.1 Content-Type: application/json

3GPP (TS 29.504)

Method	URI	Response	Status
GET	/nudr-dr/v2/subscription-data/{ueId}/authentication-data/authentication-subscription	Authentication Data (Ki, OPc, SQN)	200 OK
PATCH	/nudr-dr/v2/subscription-data/{ueId}/authentication-data/authentication-subscription	Update Authentication Data (SQN)	204 No Content
GET	/nudr-dr/v2/subscription-data/{ueId}/{servingPlmnId}/provisioned-data/am-data	AM Data	200 OK
GET	/nudr-dr/v2/subscription-data/{ueId}/{servingPlmnId}/provisioned-data/sm-data	SM Data	200 OK
GET	/nudr-dr/v2/subscription-data/{ueId}/{servingPlmnId}/provisioned-data/smf-select-data	SMF Select Data	200 OK

API (TS 29.504)

Method	URI	Response	Status
PUT	/nudr-dr/v2/subscription-data/{ueId}/context-data/amf-3gpp-access	AMF	204 No Content
PUT	/nudr-dr/v2/subscription-data/{ueId}/context-data/smf-registrations/{pduSessionId}	SMF	204 No Content

API (TS 29.519)

Method	URI	Response	Status
GET	/nudr-dr/v2/policy-data/ues/{ueId}/am-data	AM	200 OK
GET	/nudr-dr/v2/policy-data/ues/{ueId}/sm-data	SM	200 OK

4. API

OmniUDR is an Elixir application that provides a REST API for managing user data. The API is implemented using the `omniudr` library.

□□□□

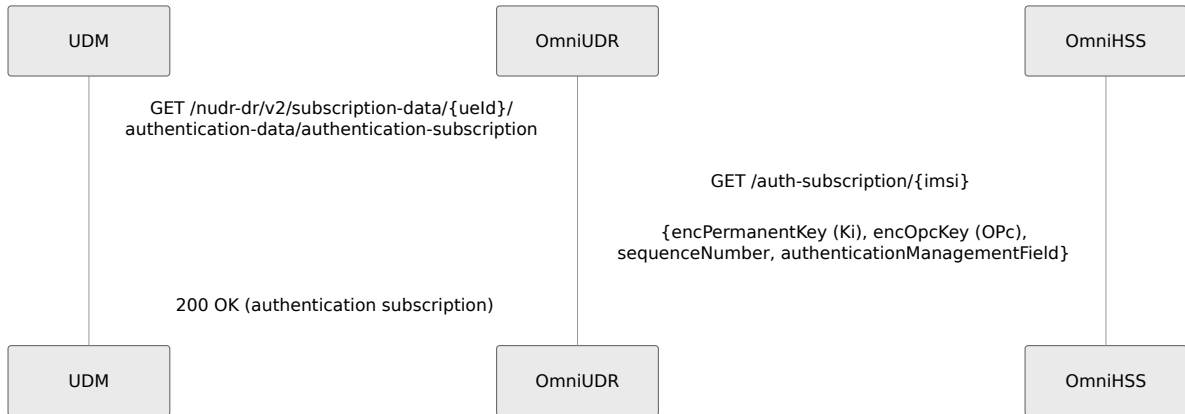
```
config :omniudr,  
  sbi_scheme: "http",  
  sbi_addr: "127.0.0.22",  
  sbi_port: 7777,  
  nrf_uri: "http://127.0.0.10:7777",  
  mcc: "999",  
  mnc: "70",  
  heartbeat_interval: 10_000,  
  hss_api_base_url: "https://127.0.0.1:8443"
```

□□□

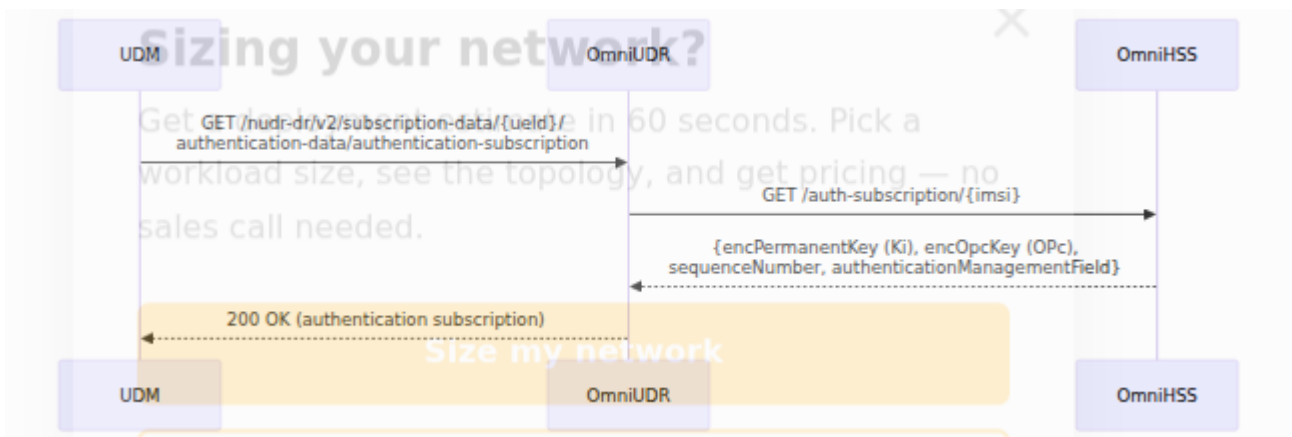
□□	□□	□□□	□□
sbi_scheme	□□ □	"http"	SBI HTTP □□□□ URI □□
sbi_addr	□□ □	"127.0.0.22"	SBI HTTP □□□□□ □ IP □□
sbi_port	□□	7777	SBI HTTP □□□□□ □ TCP □□
nrf_uri	□□ □	"http://127.0.0.10:7777"	□□ NF □□□□□□ NRF □□ URI
mcc	□□ □	"999"	□□ PLMN □□□□□ □□
mnc	□□ □	"70"	□□ PLMN □□□□□ □□
heartbeat_interval	□□ (□ □)	10000	OmniUDR □□ NRF □□ PATCH □□□□□
hss_api_base_url	□□ □	"https://127.0.0.1:8443"	OmniHSS REST API □□□ URL□□□□ □□□□□□□□□□□□

5. □□□□

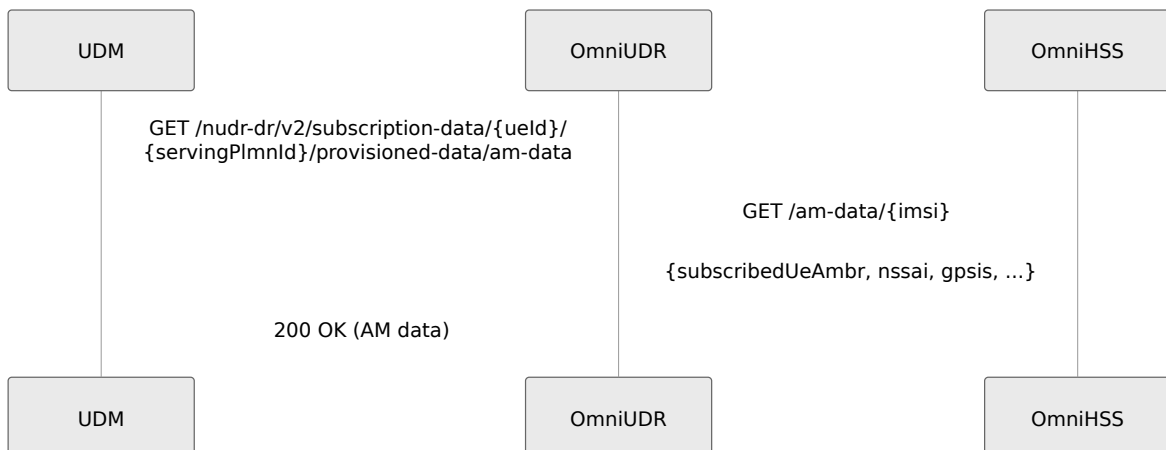
5.1 □□□□□□



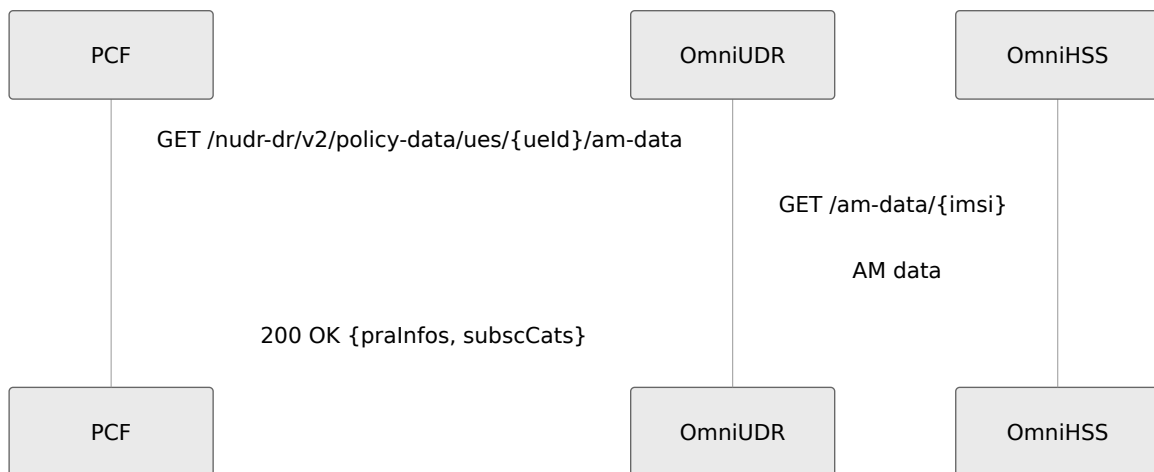
5.2 SQN □□



5.3 □□□□□□□□



5.4 序列图



6. Prometheus ☐☐

UDR ☐☐

☐☐	☐☐	☐☐	☐☐
<code>omni_udr.nrf.registration.status</code>	gauge	<code>nf_type</code>	NRF ☐☐☐ ☐ (1=☐☐ ☐, 0=☐☐ ☐)
<code>omni_udr.hss.health</code>	gauge	--	HSS ☐☐☐ ☐☐☐ (1= ☐☐, 0=☐ ☐)
<code>omni_udr.hss_requests.total</code>	counter	<code>endpoint,</code> <code>result</code>	☐ HSS ☐ ☐☐☐☐☐
<code>omni_udr.hss_request.duration_ms</code>	distribution	<code>endpoint</code>	HSS ☐☐☐ ☐☐☐☐☐☐☐ ☐☐☐☐☐☐☐ 5, 10, 25, 50, 100, 250, 500, 1000, 2500☐

BEAM VM

Key	Type	Description
<code>beam.memory.total</code>	gauge	BEAM memory usage
<code>beam.memory.processes</code>	gauge	Erlang processes memory usage
<code>beam.memory.system</code>	gauge	System memory usage (ETS, atoms, code)
<code>beam.processes.count</code>	gauge	Erlang process count
<code>beam.vm.uptime</code>	gauge	VM uptime

7. UDRs

ID	Category	Description
UDR-1	Core	OmniHSS core UDR providing AM and SM data to HSS
UDR-2	Core	Core UDR providing AM/SM data to HSS via <code>/policy-data/ues/{ueId}/am-data</code> and <code>.../sm-data</code> endpoints, compliant with TS 29.519
UDR-3	Core	Core UDR implementing <code>Nudr_DataRepository_Subscribe</code> service
UDR-4	SMF	SMF UDR implementing <code>put_smf_registration</code> service to OmniHSS
UDR-5	Core	OmniUDR core UDR providing data to OmniHSS

8. 测试

GET 返回 404 错误

OmniUDR 调用 OmniHSS

1. `hss_api_base_url` 为 OmniUDR 配置
2. 调用 IMSI 为 OmniHSS 配置
3. 返回结果 `ueId` 为 `imsi-{digits}`

PATCH SQN 返回 500

HSS 返回 SQN 失败 OmniHSS 返回 UDR 返回 `Failed to update SQN for {ueId}: {reason}` 返回 `:ok`

测试

`/policy-data/` 调用 HSS AM/SM 测试

HSS 测试

调用 `omni_udr.hss.health` gauge (1=成功, 0=失败)

`omni_udr.hss_request.duration_ms` 测试 HSS 请求