

EIR

HSS EIR EIR IMEI

- **S13** Diameter
- **IMEI** IMEI/IMEISV
- IMEI/IMEISV IMSI
-
-
- **REST API** EIR CRUD

Diameter

	ID		
S13	16,777,252	MME/SGSN	

EIR

EIR_RULE		
int	id	PK
string	action	
string	regex	
timestamp	inserted_at	
timestamp	updated_at	

動作

- whitelist - 許可
- blacklist - 拒否
- greylist - 監視

IMEI/IMEISV と IMSI

動作

動作	値	説明	備考
許可	0	許可	デフォルト
拒否	1	拒否	
監視	2	監視	

S13

動作

ECR/ECN

MME/SGSN → HSS

MMME 消息格式

AVP

- Session-Id
- Origin-Host, Origin-Realm
- Destination-Realm
- Auth-Session-State
- Terminal-Information
 - IMEI 15 字节
 - Software-Version 2 字节
- User-Name IMSI 字节
- Vendor-Specific-Application-Id

EIR

1. IMEI Software-Version 字节 IMSI 字节
2. IMSI
 - 字节
 - 字节
3. 字节
 - **IMEISV** 字节 IMEI + Software-Version 字节
 - **IMEI** 字节 IMEI
 - **IMSI** 字节
 - 字节
4. 字节

AVP

- Session-Id 字节
- Result-Code: 2001 字节
- Equipment-Status: 0 字节 / 1 字节 / 2 字节

消息

- Experimental-Result: 5422 字节 / 字节
- Experimental-Result: 5012 字节

Android

Android

EIR (Equipment Identity Register)

1. IMEISV (IMEI + Software-Version)
↓ (IMEI + Software-Version)
2. IMEI
↓ (IMEI)
3. IMSI (International Mobile Subscriber Identity)
↓ (IMSI)
4. (IMEI + IMSI)

Android

1. IMEISV

- IMEI + Software-Version "35979139461611" + "08" = "3597913946161108"
- EIR (Equipment Identity Register)
- whitelist, blacklist, greylist

2. IMEI

- IMEI "35979139461611"
- EIR (Equipment Identity Register)
-

3. IMSI (International Mobile Subscriber Identity)

- IMSI "999999876543210"
- EIR (Equipment Identity Register)
-
-

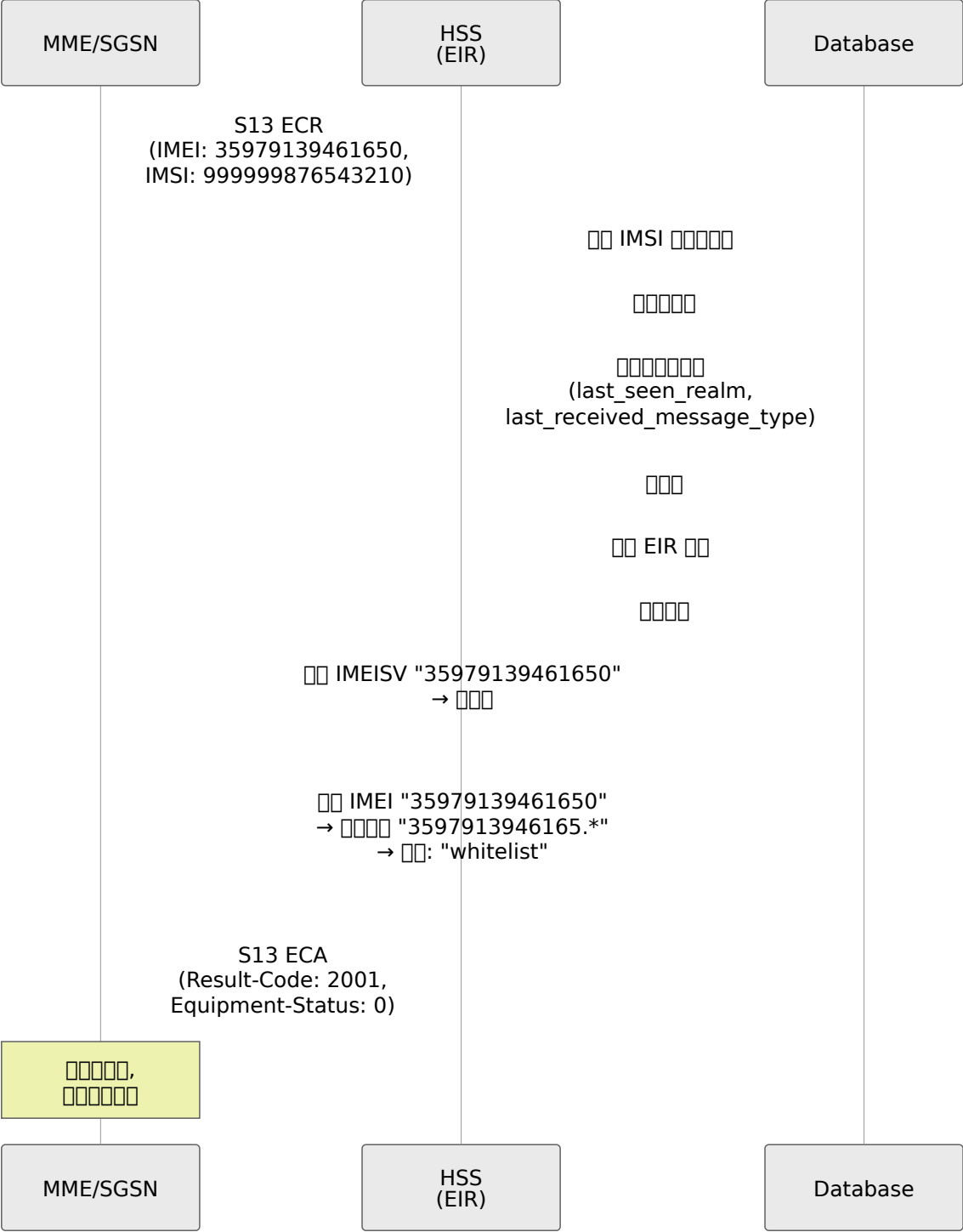
4. (IMEI + IMSI)

- `eir_unknown_equipment_behaviour`
- `:`
 - `:whitelist` -
 - `:blacklist` -
 - `:greylist` -
 - `:reject_unknown_equipment` - 5422

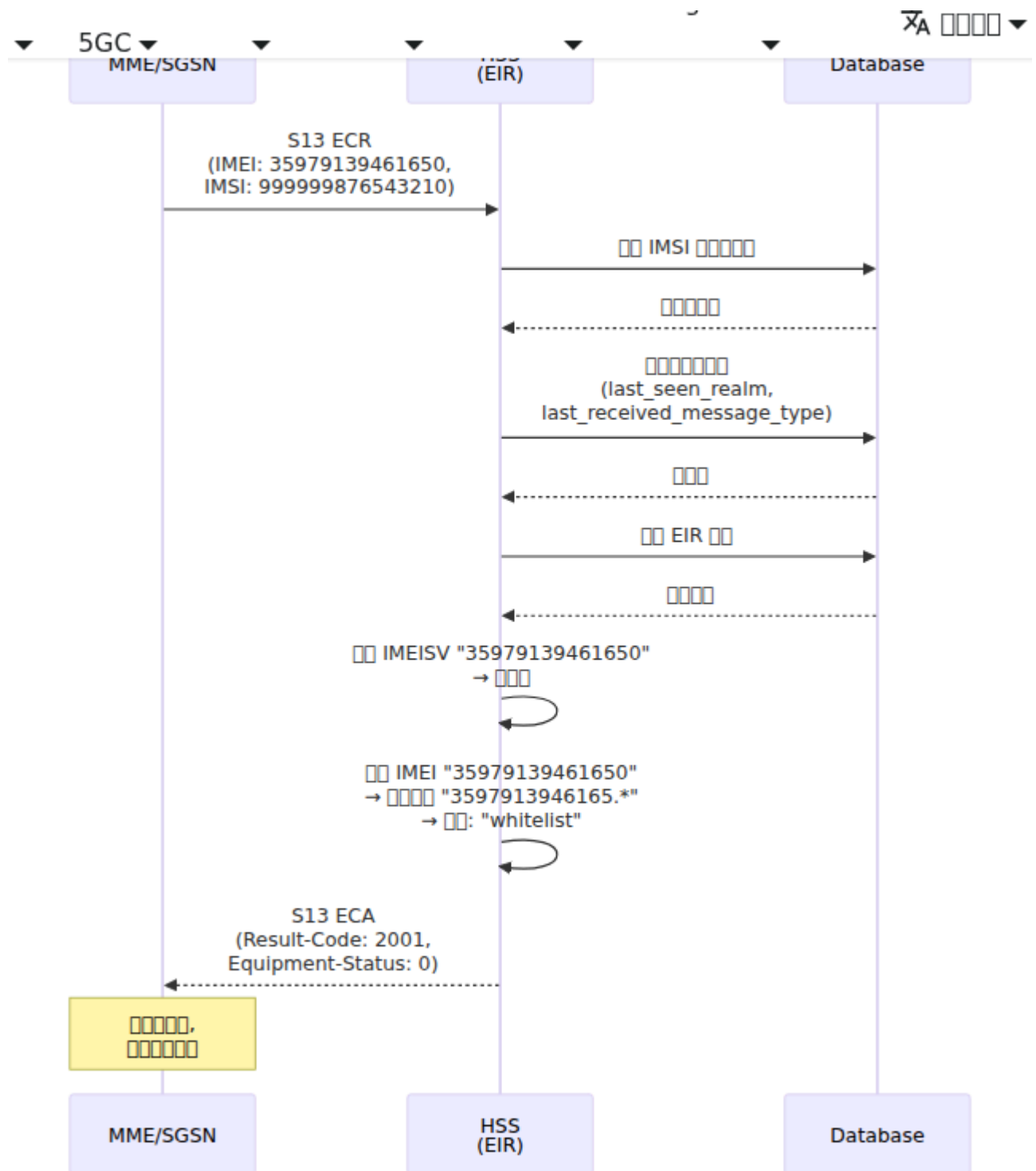
<code>"35979139461650"</code>	IMEI	/
<code>"3597913946165.*"</code>	IMEI	/
<code>"3597913946161108"</code>	IMEISV	
<code>"999999876543210"</code>	IMSI	
<code>"359791.*"</code>	TAC	

□□□□□

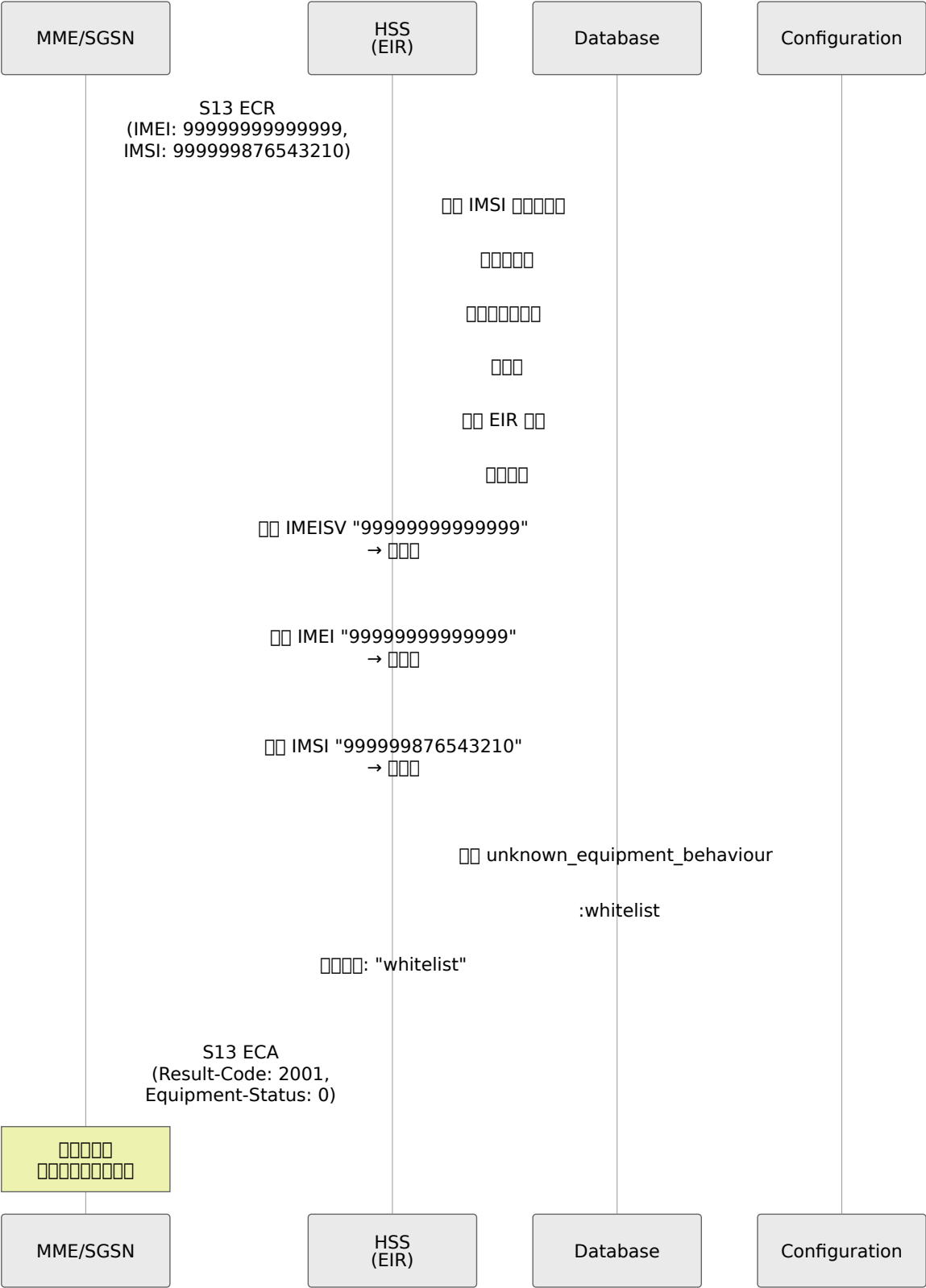
□□ 1□□□□□ - □□□□□ IMEI



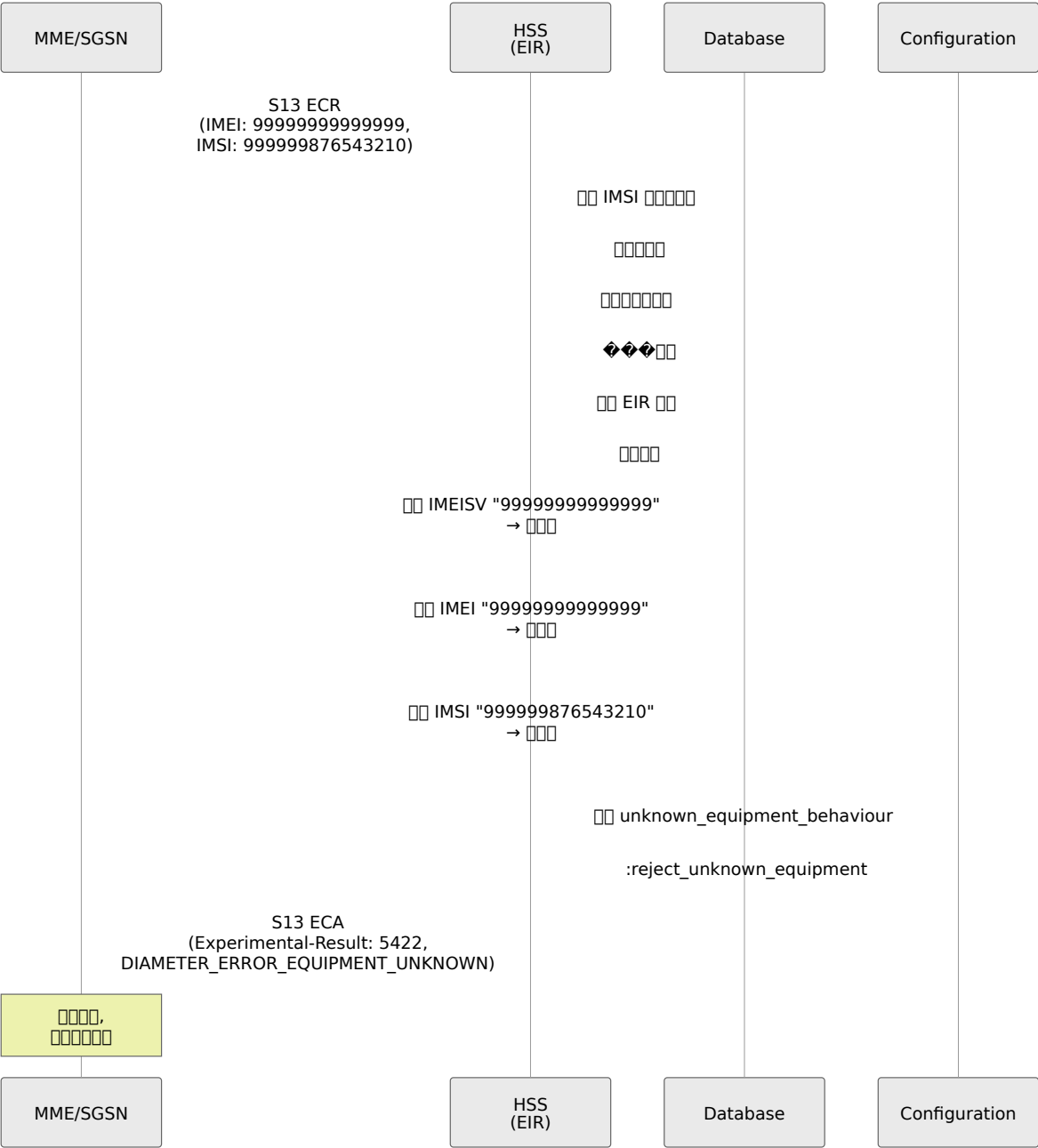
2. IMEI Whitelist



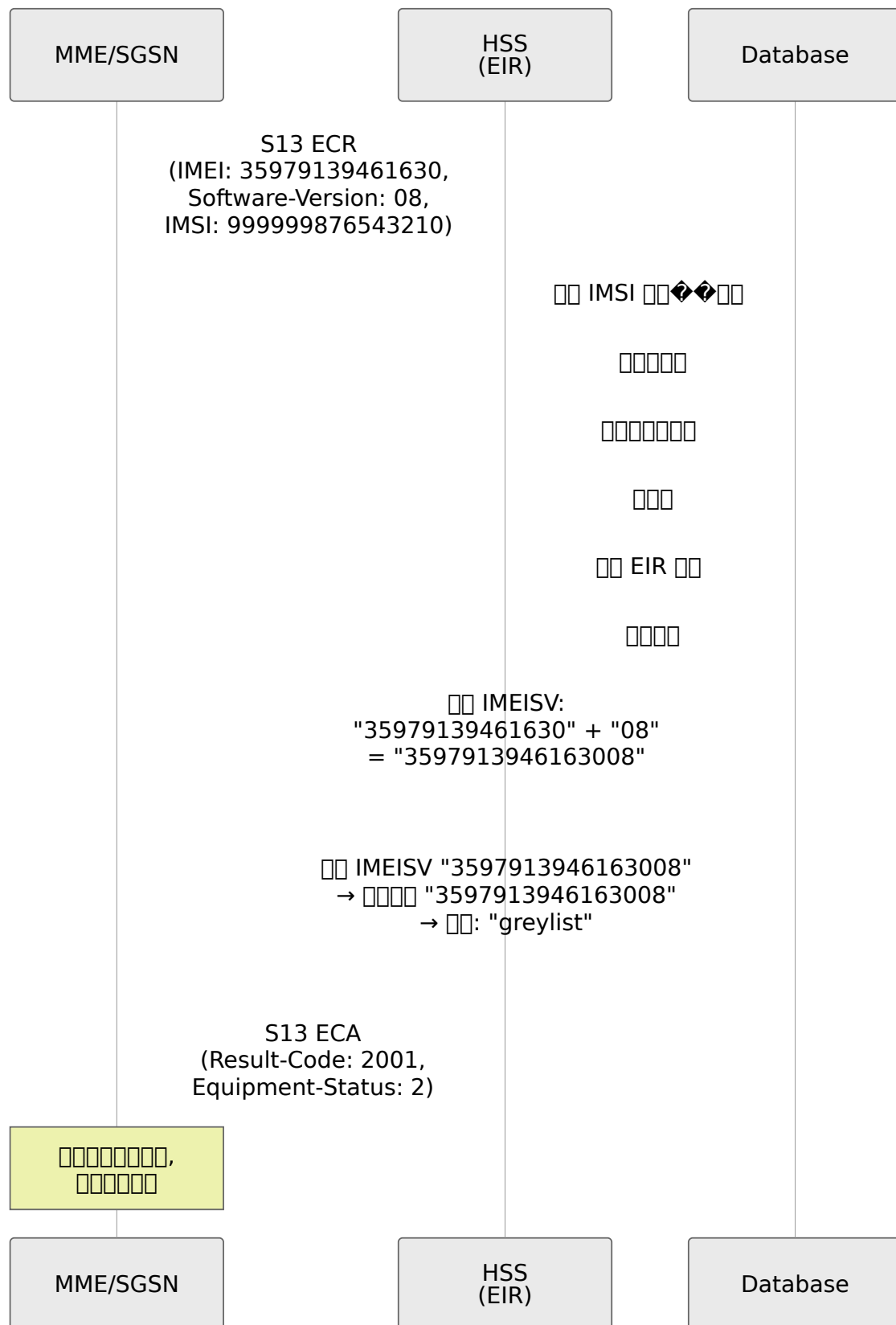
3 -



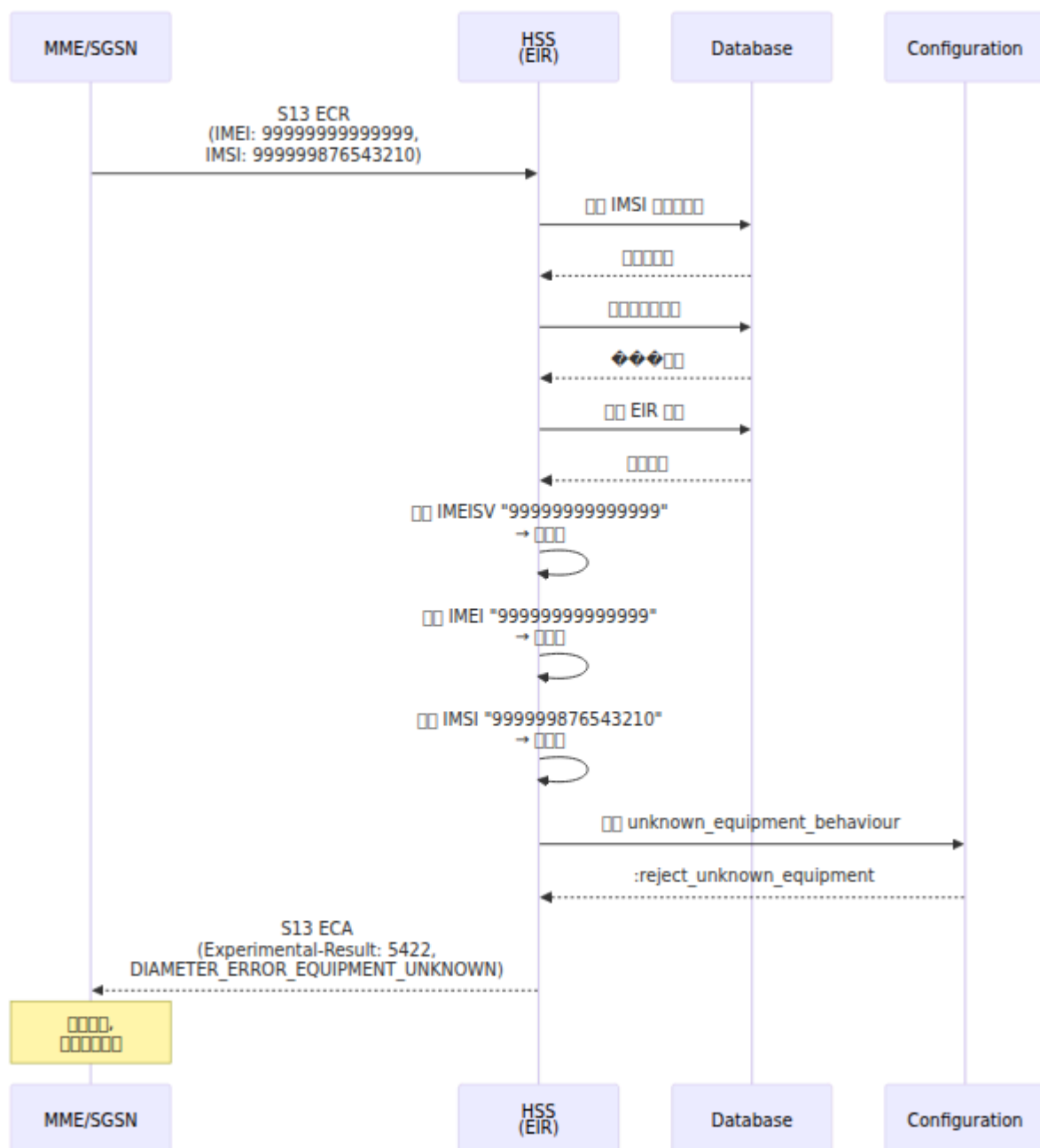
4 -



6 - IMEISV



7 - IMSI



REST API

EIR

/api/eir/rule

EIR

GET

```
GET /api/eir/rule
```

HTTP 200

```
{
  "data": [
    {
      "id": "e7f6a5b4-c3d2-4e1f-8a9b-0c1d2e3f4a5b",
      "action": "whitelist",
      "regex": "3597913946165.*",
      "inserted_at": "2025-01-15T10:30:00Z",
      "updated_at": "2025-01-15T10:30:00Z"
    },
    {
      "id": "f8a7b6c5-d4e3-4f2a-9b0c-1d2e3f4a5b6c",
      "action": "blacklist",
      "regex": "35979139461640",
      "inserted_at": "2025-01-16T14:20:00Z",
      "updated_at": "2025-01-16T14:20:00Z"
    }
  ]
}
```

EIR

GET

```
GET /api/eir/rule/{id}
```

HTTP 200

```
{
  "data": {
    "id": "e7f6a5b4-c3d2-4e1f-8a9b-0c1d2e3f4a5b",
    "action": "whitelist",
    "regex": "3597913946165.*"
  }
}
```

□□ **EIR** □□

333

```
POST /api/eir/rule
Content-Type: application/json

{
  "action": "blacklist",
  "regex": "35979139461640"
}
```

33

- `action` `whitelist` `blacklist` `greylist`
- `regex` `...` `?` `?` `...`

HTTP 201

```
{
  "data": {
    "id": "0362b7ab-9bd6-4d8a-8b1a-2f4c6e8d0a1c",
    "action": "blacklist",
    "regex": "35979139461640"
  }
}
```

HTTP 400

```
{
  "errors": {
    "regex": [""]
  }
}
```

EIR

```
PATCH /api/eir/rule/{id}
Content-Type: application/json
```

```
{
  "action": "greylist"
}
```

HTTP 200

```
{
  "data": {
    "id": "0362b7ab-9bd6-4d8a-8b1a-2f4c6e8d0a1c",
    "action": "greylist",
    "regex": "35979139461640"
  }
}
```

EIR

```
PUT /api/eir/rule/{id}
Content-Type: application/json
```

```
{
  "action": "whitelist",
  "regex": "359791394616.*"
}
```

返回HTTP 200

```
{
  "data": {
    "id": "0362b7ab-9bd6-4d8a-8b1a-2f4c6e8d0a1c",
    "action": "whitelist",
    "regex": "359791394616.*"
  }
}
```

删除 EIR

删除

```
DELETE /api/eir/rule/{id}
```

返回HTTP 204

Diameter

S13 配置 `config/runtime.exs`

```
%{
  application_name: :s13,
  application_dictionary: :diameter_gen_3gpp_s13,
  vendor_specific_application_ids: [
    %{vendor_id: 10415, auth_application_id: 16_777_252}
  ]
}
```

在 `config/runtime.exs` 中配置

000

```
config :hss, :eir,  
  unknown_equipment_behaviour: :whitelist
```

0000

- :whitelist - 0000000000000000
- :blacklist - 000000000000
- :greylist - 000000000000
- :reject_unknown_equipment - 00 Diameter 00 54220000

000

- 00/000 :whitelist - 0000000
- 00000000 :whitelist - 0000000000
- 00000000 :greylist - 000000000000
- 00000000 :reject_unknown_equipment - 00000000

00000

00000	00	00	00
2001	00	DIAMETER_SUCCESS	0000000
5422	00 0	DIAMETER_ERROR_EQUIPMENT_UNKNOWN	0000000000000000 0
5012	00 0	DIAMETER_ERROR_UNKNOWN	00000

□□

1. □□□□□□

□□□□□□□□

□□□

```
POST /api/eir/rule
{
  "action": "blacklist",
  "regex": "35979139461640" # □□ IMEI
}
```

□□□□□□□□□□□□□□□□

2. □□□□□□

□□□□□□□□□□□□

□□□

```
POST /api/eir/rule
{
  "action": "whitelist",
  "regex": "359791394.*" # □□□/□□□ TAC
}
```

□□□□□ TAC □□□□□□□□□□

3. □□□□□□□□

□□□□□□□□□□□□□□□□SIM □□

□□□

```
POST /api/eir/rule
{
  "action": "blacklist",
  "regex": "999999876543210" # IMSI
}
```

00000000 SIM 00000000000000

4.

□□□□□□□□□□

```
POST /api/eir/rule
{
  "action": "greylist",
  "regex": "35979139.*" # 35979139 TAC 35
}
```

[illegible]

5. ☐ ☐ ☐ ☐ ☐ ☐

□ □ □ □ □ □ □ □ □ □ □ □

☐ ☐ ☐

```
POST /api/eir/rule
{
  "action": "blacklist",
  "regex": "359791394616.*05" # IMEI 00 + 0000 05
}
```

IMEI 05

□□□□

□□□□

EIR □□□□□□□□□□

- **S13** □□□□ - ECR/ECA □□□
- □□□□□□ - □□□□□□□□ IMEI/IMEISV/IMSI □□
- **EIR** □□□□ - □□□□□□
- **REST API** □□□ - □□□□□□

□□□□□□□□

□□□□□□□□□□□□□□

1. **IMEISV** □□□□□□□□ IMEI + □□□□
2. **IMEI** □□□□□□ IMEI
3. **IMSI** □□□□□□ IMSI□□□□□□
4. □□□□□□□□□□□□□□

□□□□□□

- `whitelist` - □□□□□□
- `blacklist` - □□□□□□
- `greylist` - □□□□□□
- `reject_unknown_equipment` - □□□□

□□□□

IMEI □□

IMEI □□□□□□□□□□EIR□

- □□□□□□□□□□ IMEI □
- □□□□□□□□□□□□□□

- API 0000000000000000

--

EIR ID

```
00 1000 "359791.*" 00 "whitelist" 0000
00 2000 "35979139461640" 00 "blacklist" 0000
```

[illegible]

□ □ □ □ □ □ □ □ □ □ □ □ □

- 0000000000 S13 ECR 00
- REST API EIR 0000
- IMEI 0000??00000000

--	--	--	--

- Diameter 100 - S13 100mm
- API 100 - 100 API 100
- 100 - 100 HSS 100
- 100mm - 100mm

IMEI

IMEI 15

35 9791 394616 1
| | | | | Luhn
| | | | | 6
| | | | | FAC 4
| | | | | TAC 8 RBI
| | | | | RBI 2
| | | | | 6

IMEISV 16

35 9791 394616 1 08
| | | | | Luhn
| | | | | IMEI 15

IMEI/IMEISV		
359791394616108	3597913946161.*	TAC+FAC+ 359791394616*
359791394616140	35979139461614.	359791394616141-9
35979139461640	35979139461640	IMEI
3597913946163008	3597913946163008	IMEISVIMEI + SV

PCRF

HSS PCRF PCRF LTE QoS

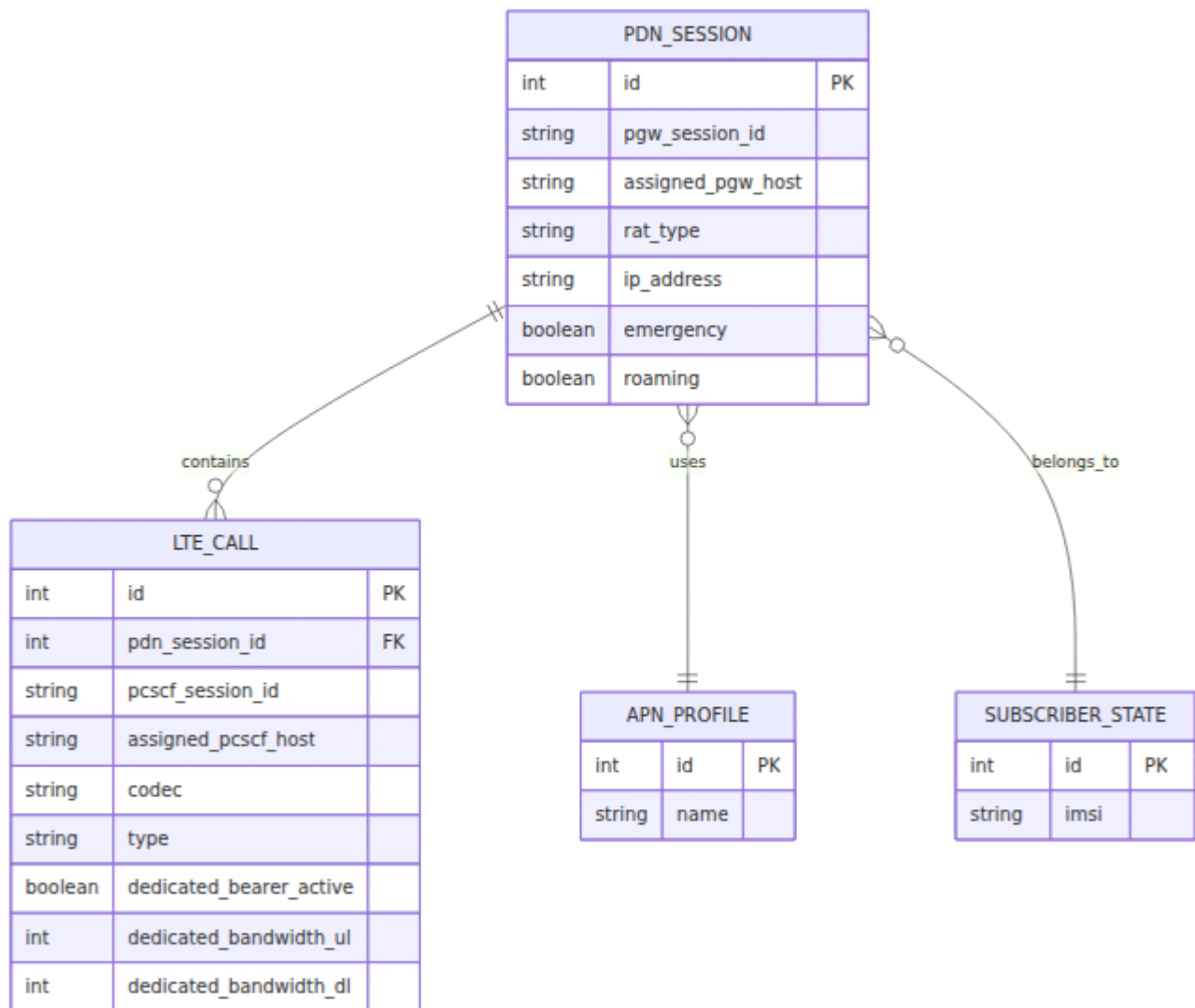
- **Gx** PGW/PCEF/
- **Rx** IMS/IP QoS
- **Sy** OCS - Sy
- RAR
- **VoLTE** QoS
- TFT
- **REST API**

Diameter

	ID		
Gx	16,777,238	PGW/PCEF	PDN QoS
Rx	16,777,236	P-CSCF/AF	IMS
Sy	16,777,302	OCS	- Sy

□□□□□□

PCRF □□□□ PDN □□□ VoLTE □□□□□□□□



Gx □□

□□□□□□

1. □□□□□□ - □□□CCR-I□

□□□PGW □□□□□□□□ PDN □□

□□ **AVPs**□

- Session-Id

- Origin-Host, Origin-Realm
- Subscription-Id[] IMSI[]
- Called-Station-Id[] APN []
- IP-CAN-Type[] IP []
- RAT-Type[]
- Framed-IP-Address[] IP []

PCRF []

1. [] IMSI []
2. [] APN [] QoS []
3. []
4. [] APN [] QoS []

[] AVPs[]

- Result-Code: 2001[] DIAMETER_SUCCESS[]
- QoS-Information[] APN []
- Default-EPS-Bearer-QoS[] QCI, ARP, []
- Bearer-Control-Mode
- []/[] - [] []/[]

2. [] - [] CCR-U[]

[] PGW [] RAT []

PCRF []

1. [] ID []
2. [] RAT []
3. []

[] Result-Code 2001[]

3. [] - [] CCR-T[]

[] PGW [] PDN []

PCRF 配置

- 1. 配置 ID 配置
- 2. 配置策略配置
- 3. 配置

配置Result-Code 2001

配置/配置

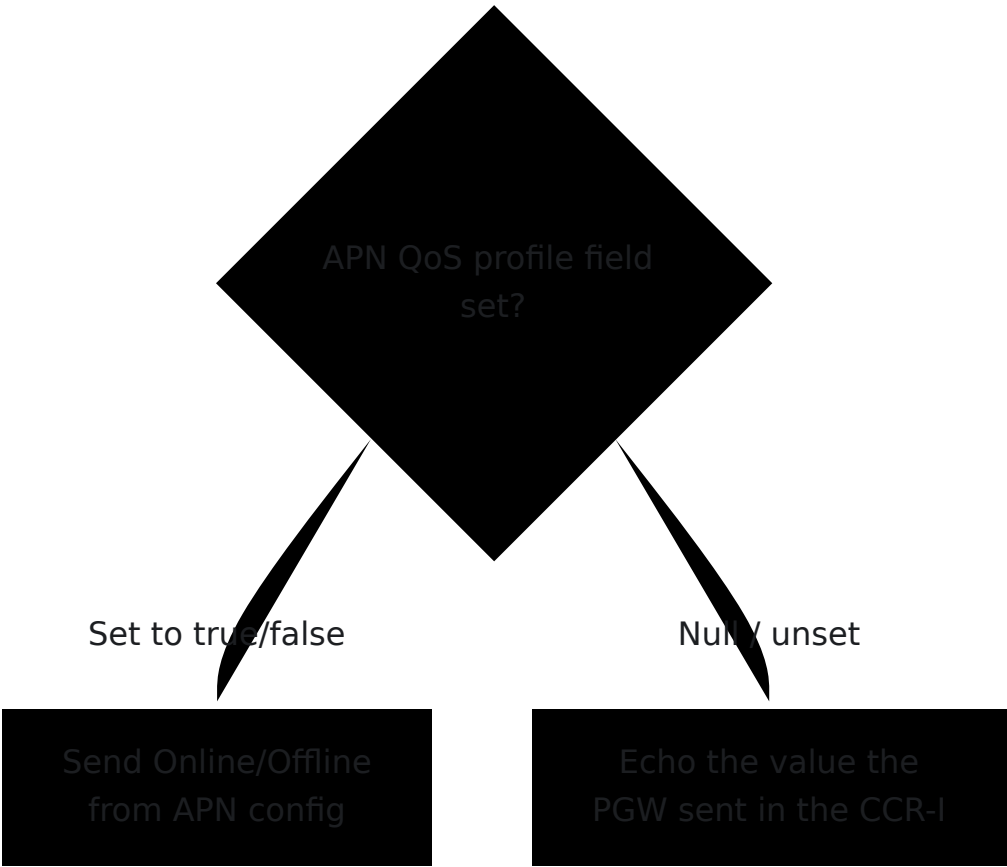
配置 Gx CCA-I 配置 Online 配置 AVP 1009 配置 Offline 配置 AVP 1008 配置 AVPs 配置 IP-CAN 配置 3GPP TS 29.212 配置 Online/Offline 配置

配置 PCRF 配置 PGW/PCEF 配置 CCR-I 配置 APN QoS 配置 配置 APN QoS 配置

配置	配置	AVP	配置
online_charging_enabled	配置	Online 配置1009	true → ENABLE_ONLINE (1) false → DISABLE_ONLINE (0)
offline_charging_enabled	配置	Offline 配置1008	true → ENABLE_OFFLINE (1) false → DISABLE_OFFLINE (0)

配置

Build Gx CCA-I



AVP TS 29.212

AVP	Code	Direction	Field Name	Description
Online	1009	0	DISABLE_ONLINE	Disable OCS / Gy charging
Online	1009	1	ENABLE_ONLINE	Enable OCS / Gy charging
Offline	1008	0	DISABLE_OFFLINE	Disable CDR / Gz charging
Offline	1008	1	ENABLE_OFFLINE	Enable CDR / Gz charging

online_charging_enabled/offline_charging_enabled fields are used to indicate the charging status of the APN. The PCRF sends the PGW the charging status of the APN in the CCR-I. The PGW then sends the charging status of the APN to the OCS / Gy or CDR / Gz.

APN OCS/APN CDR APN QoS APN

online_charging_enabled/offline_charging_enabled

Charging-Rule-Definition AVPs

4. RAR

PCRF → PGW/HSS

- IMS Rx AAR Gx RAR
- IMS Rx STR Gx RAR
- REST API

RAR AVPs

- Session-Id PGW ID
- Auth-Application-Id: 16,777,238
- Re-Auth-Request-Type 0 =
- Charging-Rule-Install/Remove
- QoS-Information

PGW OCS/HSS

PCRF TFT

- -
- -
- - QoS

- Gx RAR

- 000000000000000000000000
- 000000000005 0000000/00 IP00/0000000 TFT 00

000000

- 0000 - 0000000Spotify0WhatsApp0Facebook0000000000000000
- 000000 - 000000000000000000000000000000000000
- 00000 - 0000000000000000000000
- 000000000 - 00000000000000000000000000
- 00000 - 000000000000000000000000
- 00 **SLA** - 0000000000 QoS

QoS 00000

00000 **QoS**0000 APN 00000000

```
{
  "QoS-Class-Identifier": 9,           // QCI9 = 000000
  "APN-Aggregate-Max-Bitrate-UL": 50000, // kbps
  "APN-Aggregate-Max-Bitrate-DL": 100000, // kbps
  "Allocation-Retention-Priority": {
    "Priority-Level": 8,
    "Pre-emption-Capability": 1,       // 0000
    "Pre-emption-Vulnerability": 1    // 00000
  }
}
```

00000 **QoS**0000 VoLTE000

```
{
  "QoS-Class-Identifier": 1,           // QCI 1 = 0000
  "Max-Requested-Bandwidth-UL": 128000, // bps
  "Max-Requested-Bandwidth-DL": 128000, // bps
  "Guaranteed-Bitrate-UL": 128000,
  "Guaranteed-Bitrate-DL": 128000
}
```

Rx

1. AA AAR/ AA AAA

P-CSCF IMS VoLTE

AVPs

- Session-IdP-CSCF
- Subscription-IdIMSI SIP URI
- -
 - Max-Requested-Bandwidth-UL/DL
 - Codec-Data
 - Flow-Description5
- AF-Application-Identifier

PCRF

1. IMSI SIP URI
2. IMS
- 3.
- 4.
5. Gx RAR PGW
6. Gx RAA
7. Rx AAA

AVPs

- Result-Code: 2001 5063

SOS/ SIM

urn:service:sos SIM QoS TS 23.167

PCRF 消息交互 AAR

1. 网络侧向 PCRF 发送 Subscription-Id 消息，消息中包含 SIP URI、AAR 的 Framed-IP-Address 以及 Gx CCR-I 消息中的 PDN 名称、Gx 消息中的 APN / Called-Station-Id 以及 sos 消息中的 INVITE 消息。
2. 网络侧向 PCRF 发送 Subscription-Id 消息，消息中包含 Framed-IP-Address 以及 PDN 名称。Gx 消息中的 subscriber_state_id: nil 消息 AAR 消息。
3. **SOS APN** 消息中的 QoS 消息 SOS APN 消息“???” SOS APN 消息。
4. Gx RAR 消息中的 PDN 消息 QCI 1 消息 AAA 消息 2001 消息。

网络侧向 PCRF 发送 Framed-IP-Address 消息，Gx 消息中的 PDN 消息 UE 消息 P-CSCF 消息 AAR 消息 PCRF 消息。

网络侧向 PCRF 发送 **SOS APN** 消息，消息中包含 sos APN 消息 PCRF 消息 SOS APN 消息。

- 网络侧向 PCRF 发送 APN 消息 APN 消息 sos 消息。
- 网络侧向 PCRF 发送 OmniHSS 消息 SOS APN 消息。

网络侧向 PCRF 发送 PDN 消息 apn_profile_id: nil - 网络侧向 PCRF 发送 QoS 消息。

消息	值
网络侧向 PCRF 发送 QCI	5
网络侧向 PCRF 发送 QCI	1 ARP 消息 2
网络侧向 PCRF 发送	1
网络侧向 PCRF 发送	消息
网络侧向 PCRF 发送	消息
APN-AMBR 消息/消息	10000 kbps

0000000000 PCRF 0000000000000000000000000000000000 505 APN 0000 - 000000000000
 000000

2. 資料STR/資料STA

□□□P-CSCF □□ IMS □□□□□□□□

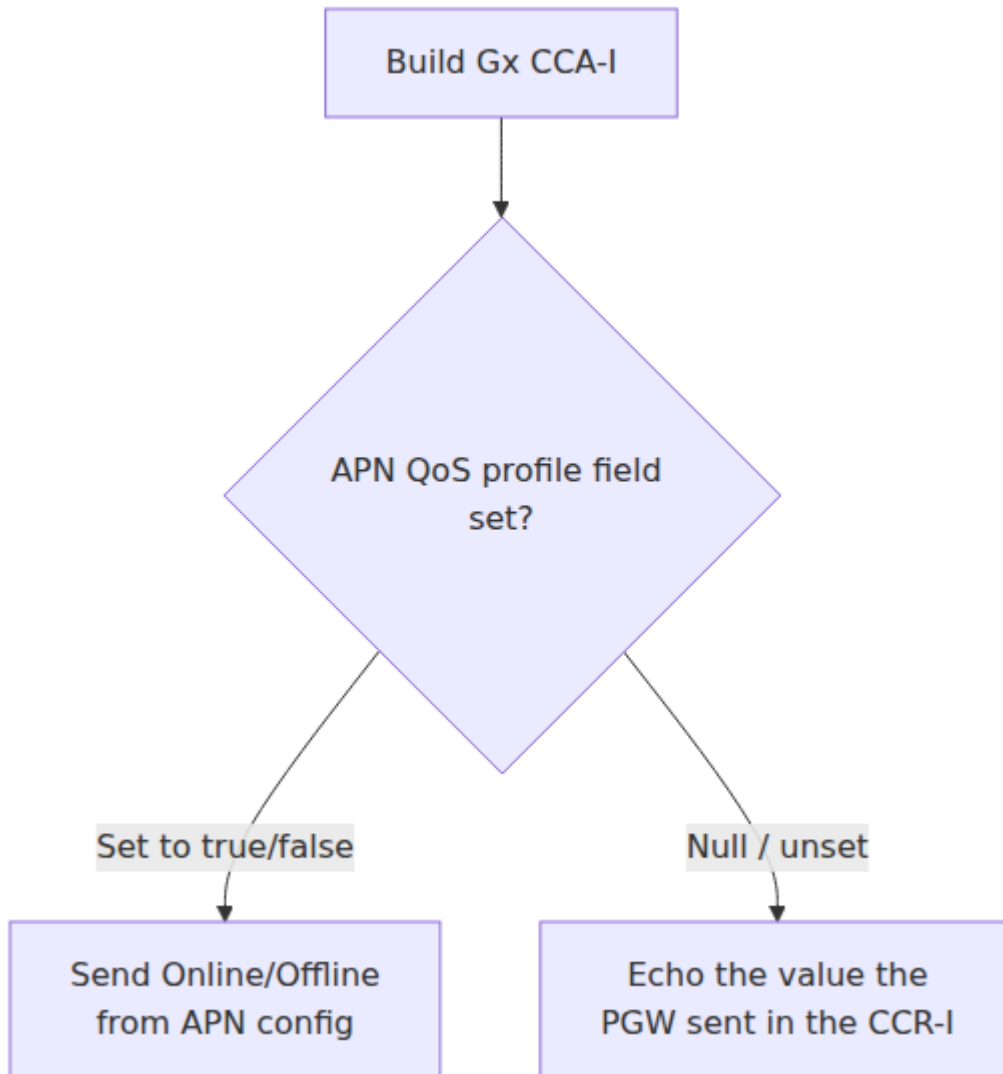
PCRF ☐☐☐

1. 取得 P-CSCF 的 IP 地址
2. 取得 **Gx RAR** 的 **PGW** IP 地址
3. 取得 IP 地址
4. 取得 STA 的 IP

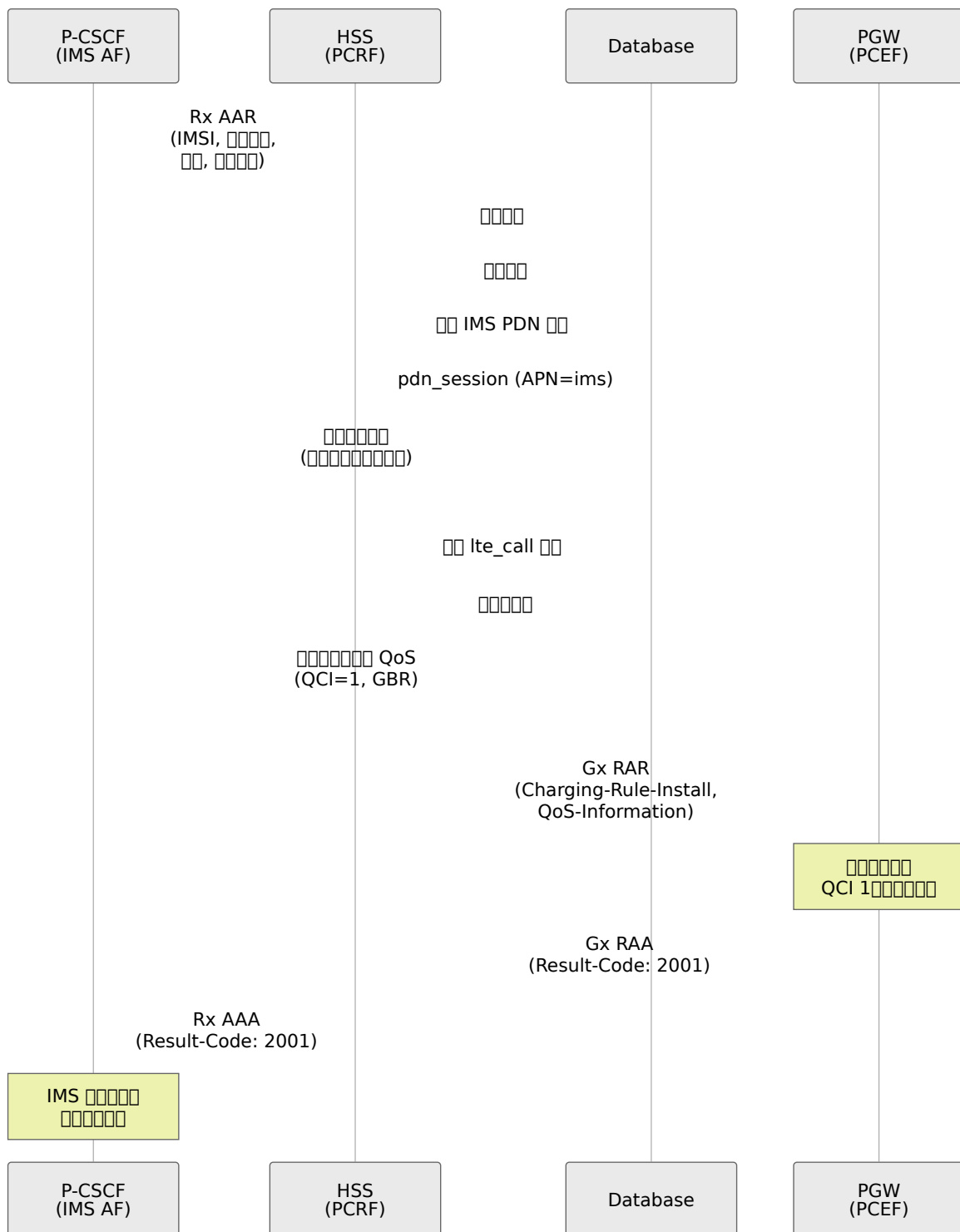
Result-Code 2001

□□□□□

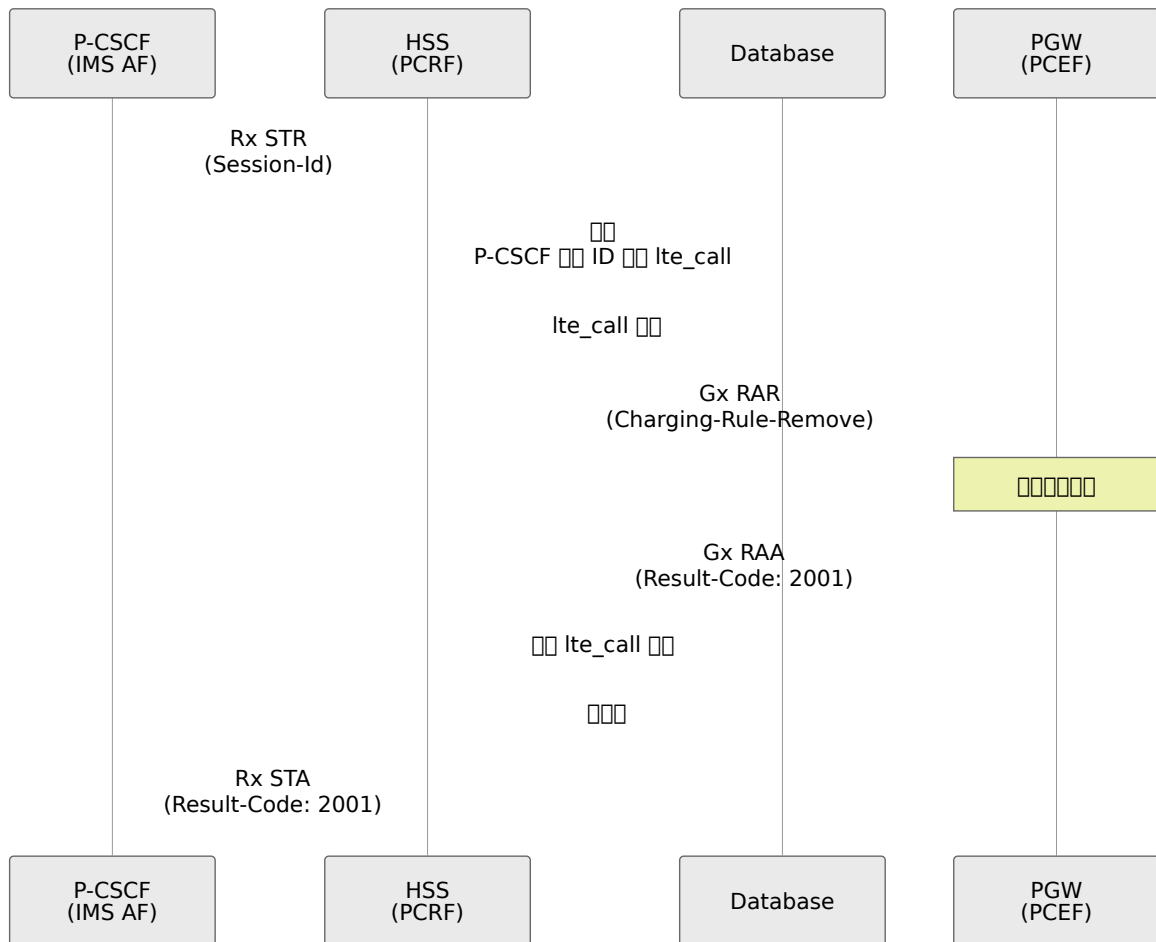
□□ **1**□**PDN** □□□□



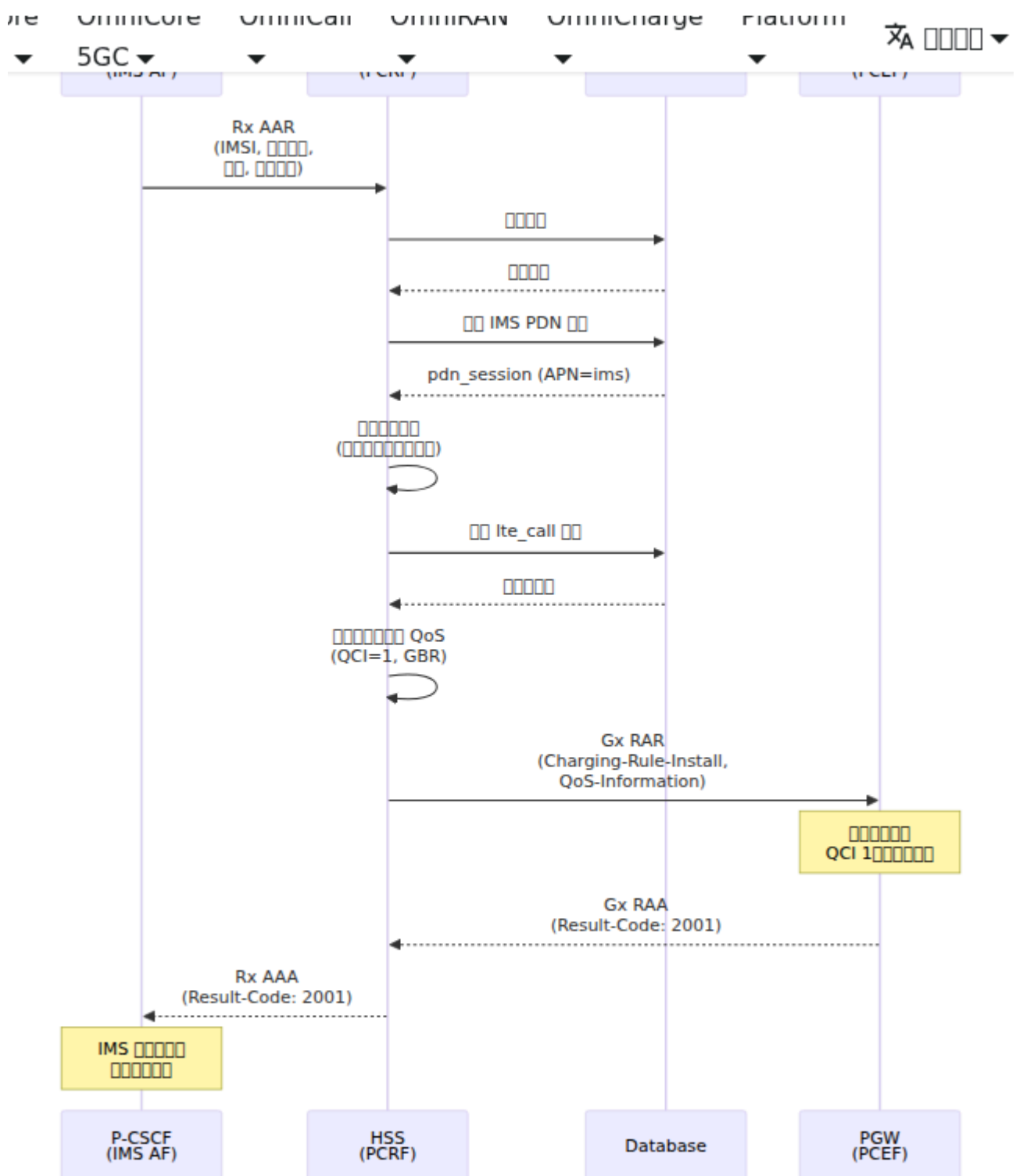
2 VoLTE Rx AAR → Gx RAR



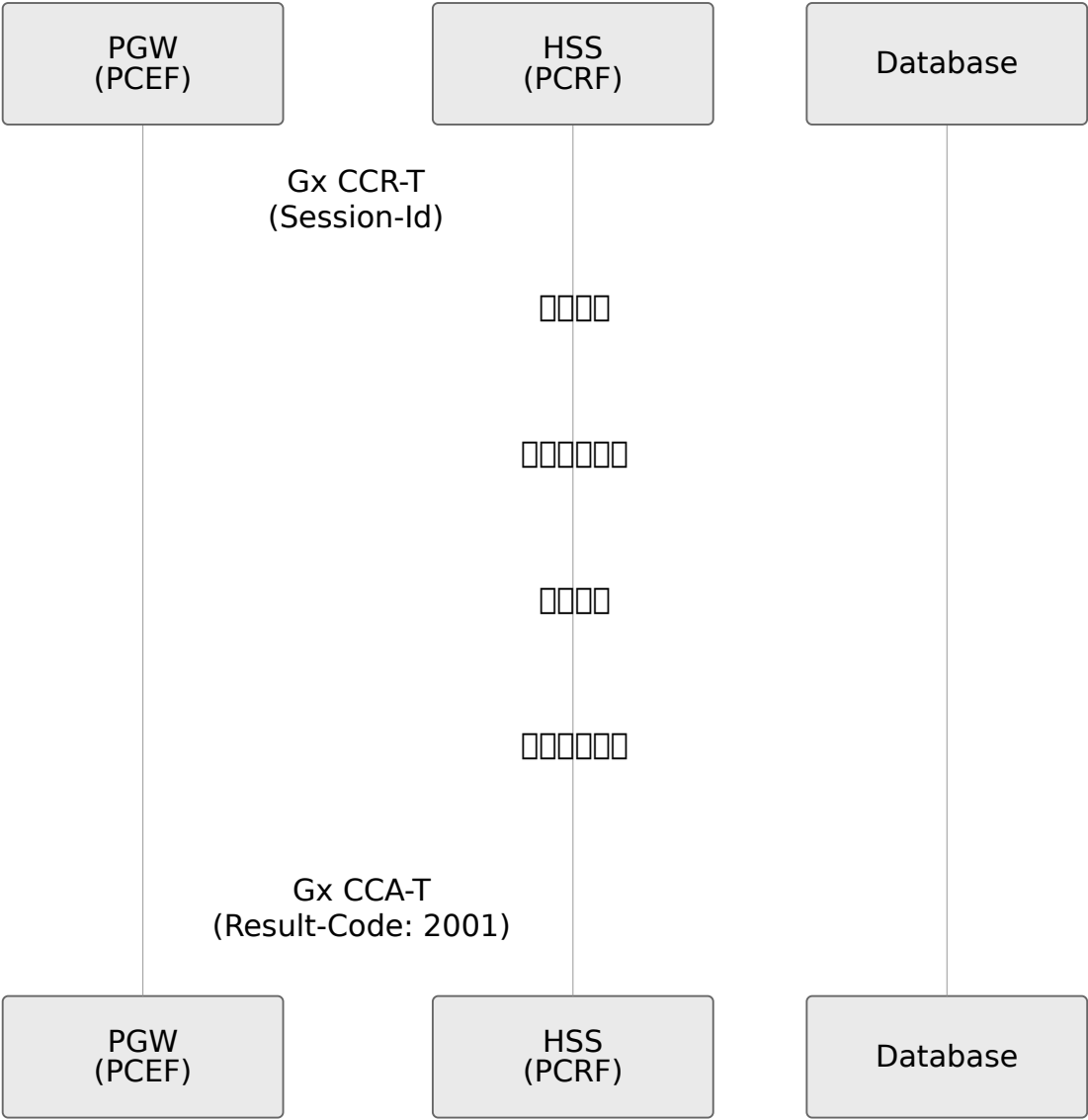
3 VoLTE Rx STR → Gx RAR



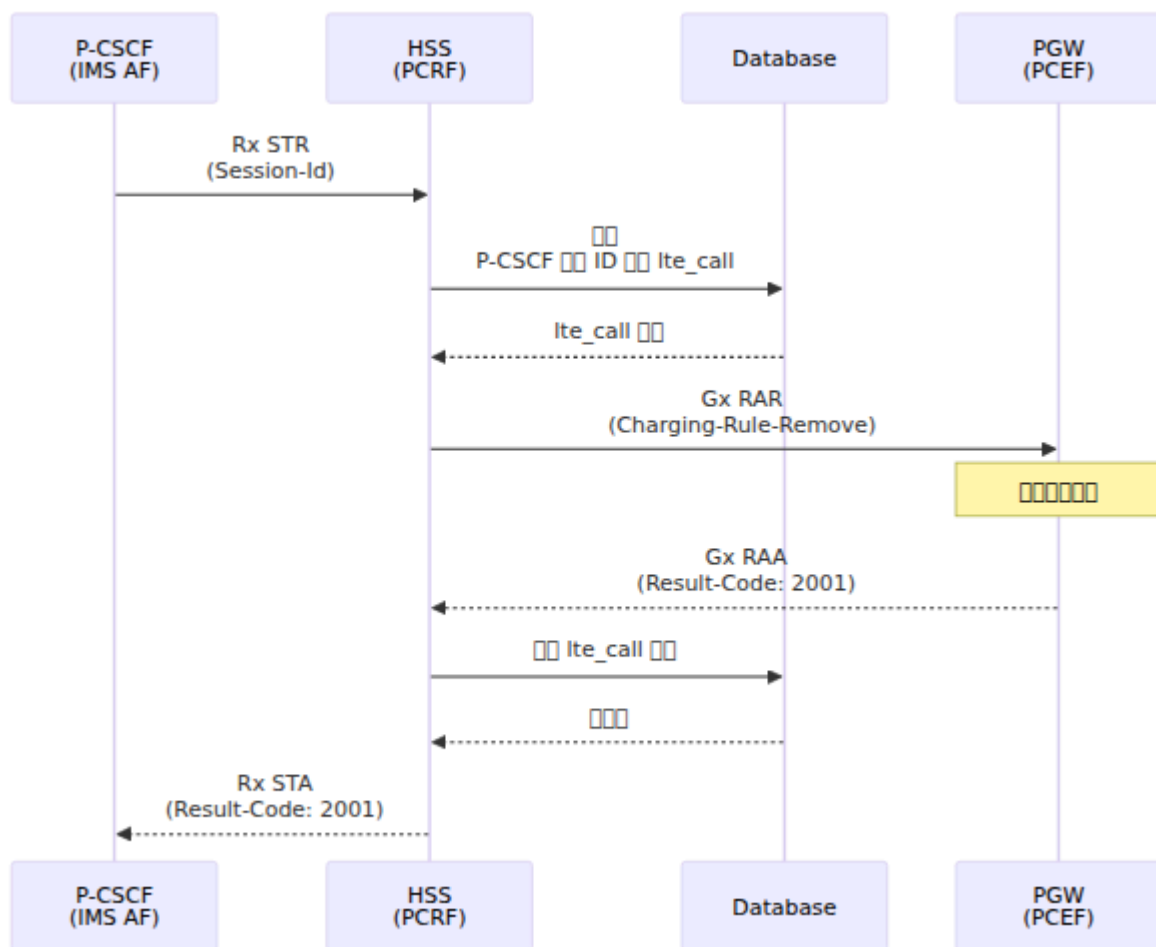
4 PDN



5 PDN



6 REST API



REST API

PCRF

POST /api/operation/pcrf_re_auth

Gx

APN QoS

```
{
  "imsi": "999999876543210",
  "apn": "ims"
}
```

成功HTTP 200

```
{
  "data": "Gx  pgw.epc.mnc999.mcc999.3gppnetwork.org
Result-Code: 2001"
}
```

失敗HTTP 400

```
{
  "error": "999999876543210  PDN "
}
```

API

PCRF APN QoS REST API

APN QoS QCI APN PDN PGW Gx RAR

```
APN  → APN QoS  → APN 
   ↓           ↓           ↓
"internet"  QCI, AMBR, ARP  
```

1. APN

APN IP

POST /api/apn/identifier

Body

```
{
  "apn_identifier": {
    "apn": "internet",
    "ip_version": "ipv4v6"
  }
}
```

IP

- "ipv4" - IPv4
- "ipv6" - IPv6
- "ipv4v6" - IPv4 or IPv6
- "ipv4_or_ipv6" - IPv4 or IPv6

Response HTTP 201

```
{
  "data": {
    "id": "4fa85f64-5717-4562-b3fc-2c963f66afa6",
    "apn": "internet",
    "ip_version": "ipv4v6"
  }
}
```

Headers

- apn 1-254
- ip_version

APN GET /api/apn/identifier

2. APN QoS

QoS QCI

POST /api/apn/qos_profile

```
{
  "apn_qos_profile": {
    "name": " ",
    "qci": 9,
    "allocation_retention_priority": 8,
    "apn_ambr_dl_kbps": 100000,
    "apn_ambr_ul_kbps": 50000,
    "pre_emption_capability": false,
    "pre_emption_vulnerability": true,
    "online_charging_enabled": false,
    "offline_charging_enabled": true
  }
}
```

QoS

項目	型	範囲	説明
name	文字列	1-254 文字	名前
qci	整数	1-254	QoS 優先度 1-4 = GBR 5-9 = GBR
allocation_retention_priority	整数	1-15	ARP 優先度 1 = 優先度
apn_ambr_dl_kbps	整数	1-4,294,967,293	APN 最大ダウンロード速度 (kbps)
apn_ambr_ul_kbps	整数	1-4,294,967,293	APN 最大アップロード速度 (kbps)
pre_emption_capability	ブール値	true/false	優先度引き上げ能力
pre_emption_vulnerability	ブール値	true/false	優先度引き下げ脆弱性
online_charging_enabled	ブール値	true/false/null	オンライン AVP Gx CCA-I null 有効 PGW CCR-I 有効 有効/無効
offline_charging_enabled	ブール値	true/false/null	オフライン AVP Gx CCA-I null 有効 PGW CCR-I 有効 有効/無効

QCI

- 1 - VoLTE - GBR 100ms
- 2 - - GBR 150ms
- 5 - IMS - GBR 100ms
- 9 - - GBR 300ms

HTTP 201

```
{
  "data": {
    "id": "8d8ac610-566d-4ef0-9c22-186b2a5ed793",
    "name": "",
    "qci": 9,
    "allocation_retention_priority": 8,
    "apn_ambr_dl_kbps": 100000,
    "apn_ambr_ul_kbps": 50000,
    "pre_emption_capability": false,
    "pre_emption_vulnerability": true,
    "online_charging_enabled": false,
    "offline_charging_enabled": true
  }
}
```

QoS GET /api/apn/qos_profile

3. APN

APN QoS

POST /api/apn/profile

```
{
  "apn_profile": {
    "name": "中国移动 APN",
    "apn_identifier_id": "4fa85f64-5717-4562-b3fc-2c963f66afa6",
    "apn_qos_profile_id": "8d8ac610-566d-4ef0-9c22-186b2a5ed793"
  }
}
```

返回

- name 中国移动 APN
- apn_identifier_id 中国移动 APN 的 ID
- apn_qos_profile_id 中国移动 APN QoS 的 ID

返回 HTTP 201

```
{
  "data": {
    "id": "a1b2c3d4-e5f6-4a3b-8c2d-1e0f9a8b7c6d",
    "name": "中国移动 APN",
    "apn_identifier_id": "4fa85f64-5717-4562-b3fc-2c963f66afa6",
    "apn_qos_profile_id": "8d8ac610-566d-4ef0-9c22-186b2a5ed793"
  }
}
```

返回

- apn_identifier_id 和 apn_qos_profile_id 是必填项
- 中国移动 APN QoS 是必填项

中国移动 APN 的 GET /api/apn/profile

返回

中国移动 1 中国移动 IMS APN 中国移动 VoLTE


```

# 1. APN
curl -X POST https://hss.example.com:8443/api/apn/identifier \
-H "Content-Type: application/json" \
-d '{
  "apn_identifier": {
    "apn": "ims",
    "ip_version": "ipv4v6"
  }
}'
# [{"data": {"id": "2c3e4f5a-6b7c-4d8e-9f0a-1b2c3d4e5f6a", ...}}]

# 2. QoS IMS
curl -X POST https://hss.example.com:8443/api/apn/qos_profile \
-H "Content-Type: application/json" \
-d '{
  "apn_qos_profile": {
    "name": "IMS QoS",
    "qci": 5,
    "allocation_retention_priority": 2,
    "apn_ambr_dl_kbps": 5000,
    "apn_ambr_ul_kbps": 5000,
    "pre_emption_capability": true,
    "pre_emption_vulnerability": false
  }
}'
# [{"data": {"id": "7b44a3e0-27cc-4b21-8d7a-3f19e5c2b8d4", ...}}]

# 3. APN
curl -X POST https://hss.example.com:8443/api/apn/profile \
-H "Content-Type: application/json" \
-d '{
  "apn_profile": {
    "name": "IMS APN",
    "apn_identifier_id": "2c3e4f5a-6b7c-4d8e-9f0a-1b2c3d4e5f6a",
    "apn_qos_profile_id": "7b44a3e0-27cc-4b21-8d7a-3f19e5c2b8d4"
  }
}'
# [{"data": {"id": "91c2f3a4-5d6e-4f7a-8b9c-0d1e2f3a4b5c", ...}}]

```

2

APN EPC API APN

API 호출

API QoS 설정

```
PATCH /api/apn/qos_profile/{id}
PUT /api/apn/qos_profile/{id}
```

API - API 호출

```
# API QoS 설정
curl -X PATCH
https://hss.example.com:8443/api/apn/qos_profile/8d8ac610-566d-
4ef0-9c22-186b2a5ed793 \
-H "Content-Type: application/json" \
-d '{
  "apn_qos_profile": {
    "apn_ambr_dl_kbps": 150000,
    "apn_ambr_ul_kbps": 75000
  }
}'
```

API 호출

1. API QoS 설정
2. API 호출 QoS 설정 APN PDN
3. API 호출 PGW Gx RAR
4. PGW QoS 설정
5. API 호출

API 호출 100 QoS 설정 "API" APN API 호출 100
API 호출 150 Mbps /75 Mbps

API 호출 APN QoS 설정 API 호출 APN PDN

API 호출

```
DELETE /api/apn/identifier/{id}
DELETE /api/apn/qos_profile/{id}
DELETE /api/apn/profile/{id}
```

□□□□□

- □□□□□ APN □□□□□□□ APN □□□□ QoS □□□□
- □□□□□□□□□□□□ APN □□□□

□□□□

□□□□□□□□ **100 Mbps** / □□ **50 Mbps**□□

```
{
  "apn_qos_profile": {
    "name": "□□□□□",
    "qci": 9,
    "allocation_retention_priority": 8,
    "apn_ambr_dl_kbps": 100000,
    "apn_ambr_ul_kbps": 50000,
    "pre_emption_capability": false,
    "pre_emption_vulnerability": true
  }
}
```

□□□□□□□□ **500 Mbps** / □□ **100 Mbps**□□

```
{
  "apn_qos_profile": {
    "name": "□□□□□",
    "qci": 8,
    "allocation_retention_priority": 5,
    "apn_ambr_dl_kbps": 500000,
    "apn_ambr_ul_kbps": 100000,
    "pre_emption_capability": true,
    "pre_emption_vulnerability": false
  }
}
```

□□□/M2M□□□□□□

```
{
  "apn_qos_profile": {
    "name": "□□□ M2M",
    "qci": 9,
    "allocation_retention_priority": 10,
    "apn_ambr_dl_kbps": 1024,
    "apn_ambr_ul_kbps": 512,
    "pre_emption_capability": false,
    "pre_emption_vulnerability": true
  }
}
```

□□□□□□□□□□□□

```
{
  "apn_qos_profile": {
    "name": "□□ APN",
    "qci": 5,
    "allocation_retention_priority": 1,
    "apn_ambr_dl_kbps": 10000,
    "apn_ambr_ul_kbps": 10000,
    "pre_emption_capability": true,
    "pre_emption_vulnerability": false
  }
}
```

□□

Diameter □□□□

Gx □□□ config/runtime.exs□□

```
%{
  application_name: :gx,
  application_dictionary: :diameter_gen_3gpp_gx,
  vendor_specific_application_ids: [
    %{vendor_id: 10415, auth_application_id: 16_777_238}
  ]
}
```

Rx `config/runtime.exs`

```
%{
  application_name: :rx,
  application_dictionary: :diameter_gen_3gpp_rx,
  vendor_specific_application_ids: [
    %{vendor_id: 10415, auth_application_id: 16_777_236}
  ]
}
```

QoS

QoS

- `apn_qos_profile` APN
 - `apn_qos_profile.qci` QoS
 - `apn_qos_profile.apn_ambr_ul_kbps`
 - `apn_qos_profile.apn_ambr_dl_kbps`
 - `apn_qos_profile.priority_level`
- Rx AAR
 - QCI
 - Max-Requested-Bandwidth AVPs
 - Flow-Description AVPs

4.1.1

消息	消息	消息	消息
2001	消息	DIAMETER_SUCCESS	消息
5001	消息	消息	IMSI 消息
5002	消息	消息	PDN 消息/消息
5063	消息	消息	IMS 消息

4.1.2

4.1.2.1

PCRF 消息

- 消息 **PDN** 消息 - 消息 APN 消息
- 消息 **VoLTE** 消息 - 消息 IMS 消息
- 消息 **QoS** 消息 - 消息 APN 消息
- 消息 - 消息

4.1.2.2

PCRF 消息

- 消息 Gx 消息 消息/消息
- 消息 **TFT** 消息 消息
- 消息
- 消息

消息 TFT 消息

□□□□

- Sy □□ - □□□□ OCS □□□□□□□□
- Diameter □□ - □□□□□□□
- API □□ - □□□ API □□
- □□ - □□ HSS □□
- □□□□ - □□□□ Diameter AVP □□□

API 入門

← API 入門

API

- API 入門
- API 入門

API 入門

400 入門

```
{  
  "error": "Invalid JSON format"  
}
```

API

- JSON 入門
- API 入門
- API 入門

404 入門

```
{  
  "error": "Resource not found"  
}
```

API

- 0000/00000/000000
- URL 00 ID 000

422 00000000

```
{
  "errors": {
    "imsi": ["has already been taken"],
    "key_set_id": ["does not exist"]
  }
}
```

000

- 0000
- 000000000
- 00000000

500 00000000

```
{
  "error": "Internal server error"
}
```

000

- 00000000
 - 0000000000
-



API Request



ore OmniCore OmniCall OmniRAN OmniCharge Platform 文A □□□□ ▼
▼ 5GC ▼ ▼ ▼ ▼ ▼

Invalid JSON

Valid

400 Bad Request

Authorized?

No

Yes

401 Unauthorized

Resource Exists?

No

Yes

Size my network

404 Not Found

Data Valid?

No

Yes

422 Validation Error

Process Request

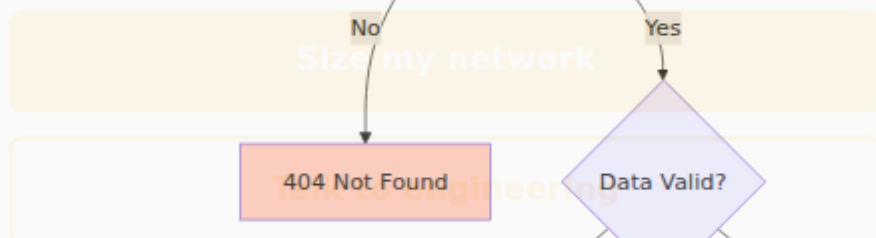
Database OK?

Error

Success

Sizing your network?

Get a deployment estimate in 60 seconds. Pick a workload size, see the topology, and get pricing — no sales call needed.





← API

API 開箱

← API 介紹

簡介

- 開箱 API
- 開箱 API
- 開箱 IP 地址

安裝與使用

安裝與使用說明書，請參閱以下連結：

安裝與使用說明書 `jq` 安裝 JSON 格式 `apt-get install jq` 或 `brew install jq` 安裝

安裝與使用

- 開箱 API
- APN 開箱
- EPC 開箱
- 開箱

1. 키셋 등록

```
KEY_SET_ID=$(curl -k -X POST
https://hss.example.com:8443/api/key_set \
-H "Content-Type: application/json" \
-d '{
  "ki": "0123456789ABCDEF0123456789ABCDEF",
  "opc": "FEDCBA9876543210FEDCBA9876543210",
  "authentication_algorithm": "milenage",
  "amf": "8000",
  "sqn": 0
}' | jq -r '.response.id')
```

2. APN QoS 등록

```
APN_QOS_ID=$(curl -k -X POST
https://hss.example.com:8443/api/apn/qos_profile \
-H "Content-Type: application/json" \
-d '{
  "name": "인터넷 QoS",
  "allocation_retention_priority": 8,
  "apn_ambr_dl_kbps": 50000,
  "apn_ambr_ul_kbps": 25000,
  "pre_emption_capability": true,
  "pre_emption_vulnerability": true,
  "qci": 9
}' | jq -r '.response.id')
```

3. APN 식별자 등록

```
APN_ID=$(curl -k -X POST
https://hss.example.com:8443/api/apn/identifier \
-H "Content-Type: application/json" \
-d '{
  "apn": "internet",
  "ip_version": "ipv4v6"
}' | jq -r '.response.id')
```

4. APN 프로파일 등록

```
APN_PROFILE_ID=$(curl -k -X POST
https://hss.example.com:8443/api/apn/profile \
-H "Content-Type: application/json" \
-d '{
  "apn_identifier_id": \"$APN_ID\",
  "apn_qos_profile_id": \"$APN_QOS_ID\",
  "name": \"인터넷 APN\"
}
```

```
}" | jq -r '.response.id')
```

5. EPC

```
EPC_PROFILE_ID=$(curl -k -X POST
https://hss.example.com:8443/api/epc/profile \
-H "Content-Type: application/json" \
-d "{
  \"apn_profiles\": [$APN_PROFILE_ID],
  \"name\": \"\",
  \"network_access_mode\": \"packet_only\",
  \"tracking_area_update_interval_seconds\": 600,
  \"ue_ambr_dl_kbps\": 100000,
  \"ue_ambr_ul_kbps\": 50000
}" | jq -r '.response.id')
```

6.

```
SUBSCRIBER_ID=$(curl -k -X POST
https://hss.example.com:8443/api/subscriber \
-H "Content-Type: application/json" \
-d "{
  \"imsi\": \"001001123456789\",
  \"key_set_id\": \"$KEY_SET_ID\",
  \"epc_profile_id\": \"$EPC_PROFILE_ID\"
}" | jq -r '.response.id')
```

```
echo "ID: $SUBSCRIBER_ID"
```

1. () -
2. (EPC) -
3. **APN** (APN) - QoS
4. () -

- MSISDN
- IMS
-

- SIM (Simulation)

111

- **MSISDN** - 11 digit number
- **IMSI** - 15 digit number

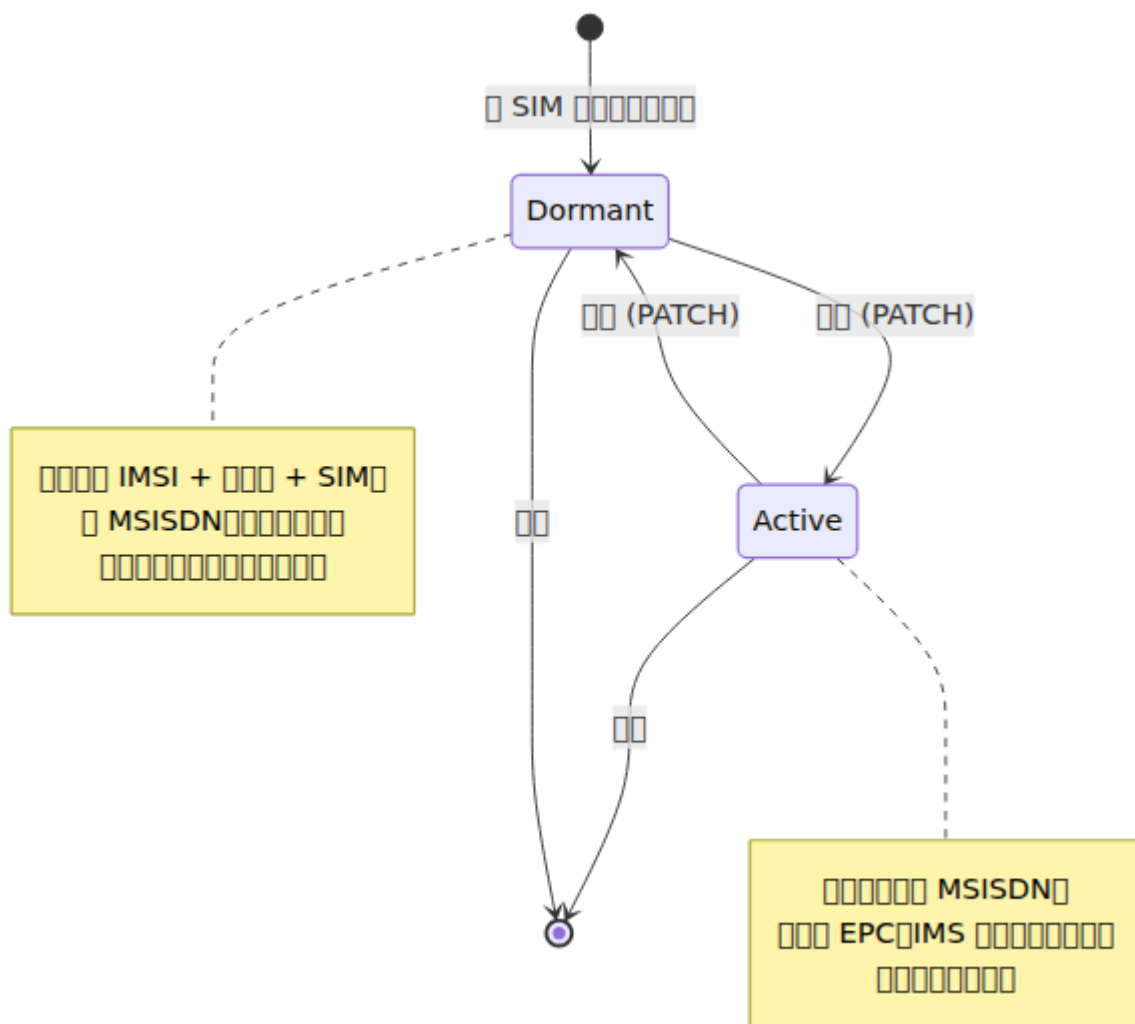
--	--	--	--	--	--	--

SIM

SIM SIM
 ICCID PIN PUK ADM Ki OPc

IMSI 國際移動用戶識別碼
 EPC 歐洲電信標準協會
 MSISDN 行動電話號碼
 SIM 智慧卡
 SIM 卡

□ □ □ □ □ □



1 SIM

□ SIM □□□□□□□□□□□□□□□□ SIM □□□□□□□□□□□□ □□□□□□□□□□□□

```
# 1. SIM 키 (Ki, OPc)
KEY_SET_ID=$(curl -k -X POST
https://hss.example.com:8443/api/key_set \
-H "Content-Type: application/json" \
-d '{
  "ki": "0123456789ABCDEF0123456789ABCDEF",
  "opc": "FEDCBA9876543210FEDCBA9876543210",
  "authentication_algorithm": "milenege",
  "amf": "8000",
  "sqn": 0
}' | jq -r '.response.id')

# 2. SIM ICCID (ICCID, PIN/PUK/ADM)
SIM_ID=$(curl -k -X POST https://hss.example.com:8443/api/sim \
-H "Content-Type: application/json" \
-d '{
  "iccid": "89440000000000000001",
  "is_esim": false,
  "ind_length": 5
}' | jq -r '.response.id')

# 3. IMSI
# imsi key_set_id SIM MSISDN
SUBSCRIBER_ID=$(curl -k -X POST
https://hss.example.com:8443/api/subscriber \
-H "Content-Type: application/json" \
-d "{
  \"imsi\": \"001010000000001\",
  \"key_set_id\": \"\$KEY_SET_ID\",
  \"sim_id\": \"\$SIM_ID\",
  \"enabled\": false,
  \"ims_enabled\": false
}" | jq -r '.response.id')
```

이제 HSS에 IMSI, KEY_SET_ID, SIM_ID, MSISDN을 등록하고 enabled를 true로 설정합니다.

2. IMSI, KEY_SET_ID, SIM_ID, MSISDN 등록

이제 HSS에 IMSI, KEY_SET_ID, SIM_ID, MSISDN을 등록하고 enabled를 true로 설정합니다.

```
curl -k -X PATCH
https://hss.example.com:8443/api/subscriber/$SUBSCRIBER_ID \
-H "Content-Type: application/json" \
-d "{
  \"enabled\": true,
  \"ims_enabled\": true,
  \"msisdns\": [\"$MSISDN_ID\"],
  \"epc_profile_id\": \"$EPC_PROFILE_ID\",
  \"ims_profile_id\": \"$IMS_PROFILE_ID\",
  \"roaming_profile_id\": \"$ROAMING_PROFILE_ID\"
}"
```

PATCH IMSI SIM MSISDN IMSI MSISDN IMSI MSISDN IMSI MSISDN

3

enabled false

```
curl -k -X PATCH
https://hss.example.com:8443/api/subscriber/$SUBSCRIBER_ID \
-H "Content-Type: application/json" \
-d '{
  "enabled": false,
  "ims_enabled": false
}'
```

IMS ims_enabled false enabled true

HSS HSS

- enabled false
- IMS ims_enabled false
- LTE EPC IMS IMS

DIAMETER_ERROR_USER_UNKNOWN (5001) S6a Cx 3GPP TS 29.272 3GPP TS 29.229 IMSI IMSI

IP

IP

APN IPv4 IoT

```
# 安装jq (apt-get install jq || brew install jq)
```

```
# 1. 创建key
```

```
KEY_SET_ID=$(curl -k -X POST
https://hss.example.com:8443/api/key_set \
-H "Content-Type: application/json" \
-d '{
  "ki": "0123456789ABCDEF0123456789ABCDEF",
  "opc": "FEDCBA9876543210FEDCBA9876543210",
  "authentication_algorithm": "milenage",
  "amf": "8000",
  "sqn": 0
}' | jq -r '.response.id')
```

```
# 2. 创建APN QoS 配置
```

```
APN_QOS_ID=$(curl -k -X POST
https://hss.example.com:8443/api/apn/qos_profile \
-H "Content-Type: application/json" \
-d '{
  "name": "IoT 配置",
  "allocation_retention_priority": 8,
  "apn_ambr_dl_kbps": 10000,
  "apn_ambr_ul_kbps": 5000,
  "pre_emption_capability": false,
  "pre_emption_vulnerability": false,
  "qci": 9
}' | jq -r '.response.id')
```

```
# 3. 创建APN 标识
```

```
APN_ID=$(curl -k -X POST
https://hss.example.com:8443/api/apn/identifier \
-H "Content-Type: application/json" \
-d '{
  "apn": "internet",
  "ip_version": "ipv4"
}' | jq -r '.response.id')
```

```
# 4. 创建APN 配置
```

```
APN_PROFILE_ID=$(curl -k -X POST
https://hss.example.com:8443/api/apn/profile \
-H "Content-Type: application/json" \
-d "{
  \"apn_identifier_id\": \"$APN_ID\",
```

```
\ "apn_qos_profile_id\ ": \ "$APN_QOS_ID\ ",  
\ "name\ ": \ "IoT \ \ \ \ APN\  
}" | jq -r '.response.id')
```

5. \ APN \ \ \ \ IP

```
STATIC_IP_ID=$(curl -k -X POST  
https://hss.example.com:8443/api/epc/static_ip \  
-H "Content-Type: application/json" \  
-d "{  
  \ "apn_profile_id\ ": \ "$APN_PROFILE_ID\  
  \ "ipv4_static_ip\ ": \ "100.64.1.100\  
}" | jq -r '.response.id')
```

6. \ \ EPC \ \ \ \

```
EPC_PROFILE_ID=$(curl -k -X POST  
https://hss.example.com:8443/api/epc/profile \  
-H "Content-Type: application/json" \  
-d "{  
  \ "apn_profiles\ ": [$APN_PROFILE_ID],  
  \ "name\ ": \ "IoT \ \ \ \ \  
  \ "network_access_mode\ ": \ "packet_only\  
  \ "tracking_area_update_interval_seconds\ ": 600,  
  \ "ue_ambr_dl_kbps\ ": 10000,  
  \ "ue_ambr_ul_kbps\ ": 5000  
}" | jq -r '.response.id')
```

7. \ \ MSISDN \ \ \ \ \ \

```
MSISDN_ID=$(curl -k -X POST  
https://hss.example.com:8443/api/msisdns \  
-H "Content-Type: application/json" \  
-d '{  
  "msisdns": "14155551000"  
}' | jq -r '.response.id')
```

8. \ \ \ \ \ \ IP \ \ \

```
SUBSCRIBER_ID=$(curl -k -X POST  
https://hss.example.com:8443/api/subscriber \  
-H "Content-Type: application/json" \  
-d "{  
  \ "imsi\ ": \ "0010019999999999\  
  \ "key_set_id\ ": \ "$KEY_SET_ID\  
  \ "epc_profile_id\ ": \ "$EPC_PROFILE_ID\  
  \ "msisdns\ ": [$MSISDN_ID],  
  \ "static_ips\ ": [$STATIC_IP_ID]
```

```
} " | jq -r '.response.id')
```

```
echo "IoT 物联网"  
echo "  ID: $SUBSCRIBER_ID"  
echo "  IMSI: 001001999999999"  
echo "  MSISDN: 14155551000"  
echo "  IPv4: 100.64.1.100  'internet' APN  "
```

物联网

物联网物联网 IoT 物联网

1. 物联网 (物联网) - 物联网
2. **APN** (APN 物联网) - “物联网”
3. **IP** (IP IP) - IPv4 100.64.1.100
4. 物联网 (EPC 物联网) - IoT 物联网
5. 物联网 (MSISDN) - 物联网
6. 物联网 (IP) - 物联网

物联网

物联网物联网“物联网” APN 物联网 IP 100.64.1.100 物联网 DHCP 物联网

物联网

- 物联网 IP 物联网 APN 物联网 APN 物联网 2-5
- 物联网物联网 IMS 物联网
- 物联网物联网 物联网
- 物联网 SIM 物联网 SIM

物联网

- IP IP - 物联网 IP
- 物联网 - 物联网 IP
- MSISDN - 物联网

← API

OmniHSS API

←

- API
-
-
- Sh RepositoryData
- MSISDN
- SIM
-
-
- IP
-
- EIR
-
-
- API

API

URL

https://[hostname]:8443/api

□□□□

- **Content-Type:** application/json
- □□: □□□ HTTPS
- □□: 8443□□□□□

□□: □□ API □□□□□□“□□”□ JSON □□□□□□□□□□

□□□□:

```
{
  "name": "value",
  "field": "value"
}
```

□□□□□□□□□:

```
{
  "subscriber": {
    "name": "value",
    "field": "value"
  }
}
```

□□:

```
# ✓ □□
curl -X POST https://hss.example.com:8443/api/ims/profile \
  -H "Content-Type: application/json" \
  -d '{"name": "default", "ifc_template": "..."}'

# x □□
curl -X POST https://hss.example.com:8443/api/ims/profile \
  -H "Content-Type: application/json" \
  -d '{"ims_profile": {"name": "default", "ifc_template": "..."}'}
```

API

API returns JSON response

Success:

```
{
  "status": "success",
  "response": { ... }
}
```

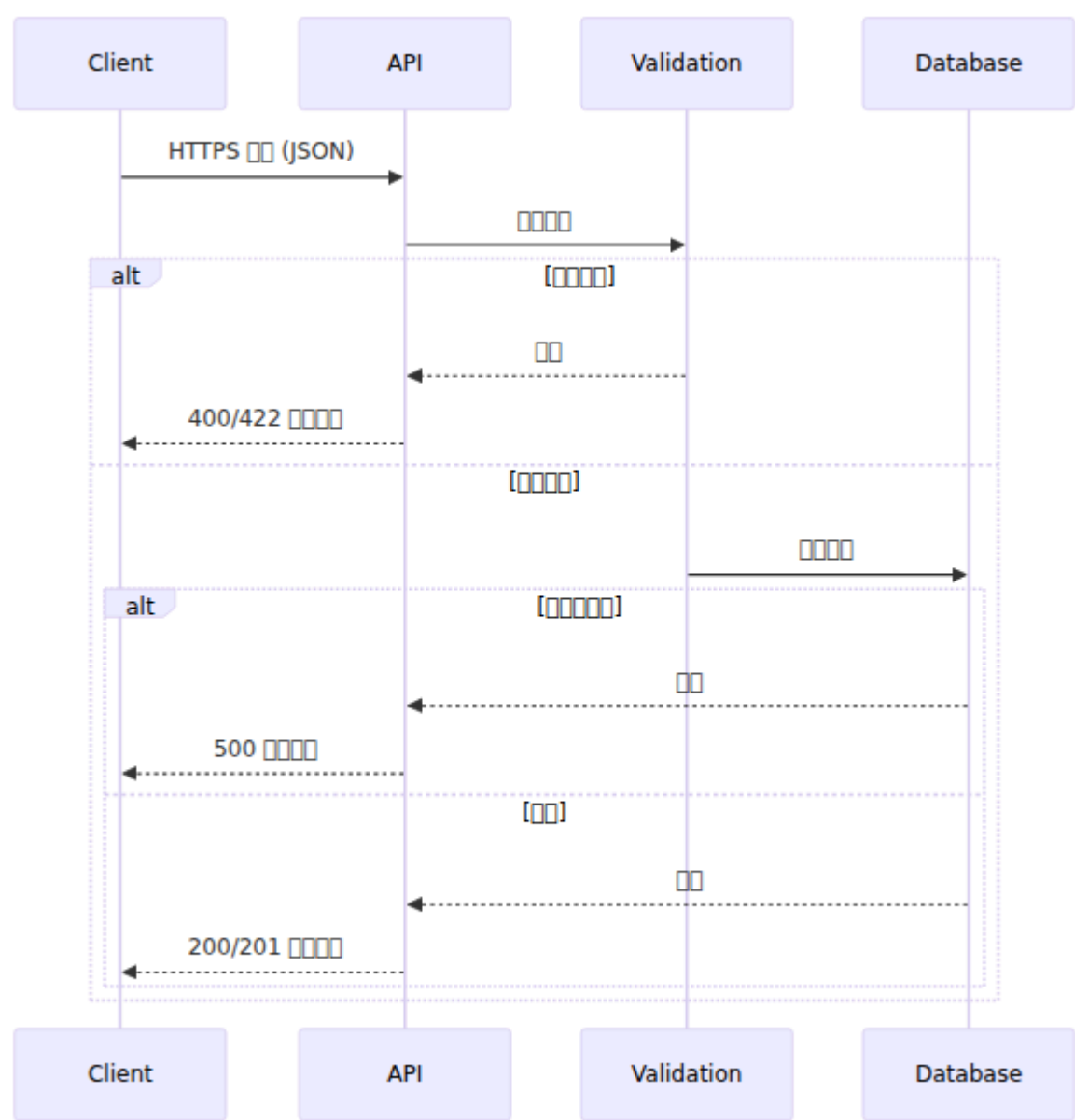
Error:

```
{
  "status": "error",
  "response": {
    "invalid_fields": {
      "field_name": "error message"
    }
  }
}
```

HTTP status

Code	Message	Method
200	OK	GET, PUT, DELETE
201	Created	POST
400	Bad Request	Invalid request
404	Not Found	Resource not found
422	Unprocessable Entity	Invalid data
500	Internal Server Error	Server error

API 流程图



流程图

流程图

流程图

请求: GET /api/subscriber

响应:

項目	型別	説明
enabled	boolean	有効/無効
ims_enabled	boolean	IMS 有効/無効

リクエスト:

```
curl -k https://hss.example.com:8443/api/subscriber
```

レスポンス:

```
{
  "data": [
    {
      "id": "d290f1ee-6c54-4b01-90e6-d701748f0851",
      "imsi": "001001123456789",
      "enabled": true,
      "ims_enabled": true,
      "sim_id": "3f2504e0-4f89-41d3-9a0c-0305e82c3301",
      "key_set_id": "6ba7b810-9dad-41d1-80b4-00c04fd430c8",
      "epc_profile_id": "e58ed763-928c-4155-bee9-fdbaaadc15f3",
      "ims_profile_id": "7c9e6679-7425-40de-944b-e07fc1f90ae7",
      "roaming_profile_id": "9c858901-8a57-4791-81fe-4c455b099bc9",
      "custom_attributes": {},
      "inserted_at": "2025-10-15T10:30:00Z",
      "updated_at": "2025-10-15T10:30:00Z"
    }
  ]
}
```

サブスクリバ ID 取得

サブスクリバ ID 一覧取得

リクエスト: GET /api/subscriber/:id

요청:

필드	타입	설명
id	string (UUID)	유니크 ID

요청:

```
curl -k https://hss.example.com:8443/api/subscriber/d290f1ee-6c54-4b01-90e6-d701748f0851
```

IMSI 조회

IMSI 조회

요청: GET /api/subscriber/imsi/:imsi

요청:

필드	타입	길이	설명
imsi	string	14-15	14-15 자리

요청:

```
curl -k https://hss.example.com:8443/api/subscriber/imsi/001001123456789
```

요청: IMSI 조회

MSISDN 조회

MSISDN 조회

요청: GET /api/subscriber/msisdn/:msisdn

項目:

項目	型	説明	制約
msisdn	string	ISDN 番号	1-15 桁 (E.164)

リクエスト:

```
curl -k
https://hss.example.com:8443/api/subscriber/msisdn/14155551234
```

レスポンス: 200 OK

レスポンス

レスポンス

メソッド: POST /api/subscriber

レスポンス:

```
{
  "subscriber": {
    "imsi": "001001123456789",
    "enabled": true,
    "ims_enabled": true,
    "sim_id": "3f2504e0-4f89-41d3-9a0c-0305e82c3301",
    "key_set_id": "6ba7b810-9dad-41d1-80b4-00c04fd430c8",
    "epc_profile_id": "e58ed763-928c-4155-bee9-fdbaaadc15f3",
    "ims_profile_id": "7c9e6679-7425-40de-944b-e07fc1f90ae7",
    "roaming_profile_id": "9c858901-8a57-4791-81fe-4c455b099bc9",
    "custom_attributes": {
      "note": "テスト"
    }
  }
}
```

項目:

- `imsi` - 14-15 位元組
- `key_set_id` - 32 位元組
- `epc_profile_id` - EPC 32 位元組

選擇:

- `enabled` - 是否: true
- `ims_enabled` - 是否: true
- `sim_id` - SIM 卡
- `ims_profile_id` - IMS 32 位元組 IMS 32 位元組
- `roaming_profile_id` - 32 位元組 32 位元組 32 位元組
- `msisdns` - MSISDN ID 32 位元組
- `static_ips` - APN 32 位元組 IP ID 32 位元組
- `custom_attributes` - 32 位元組

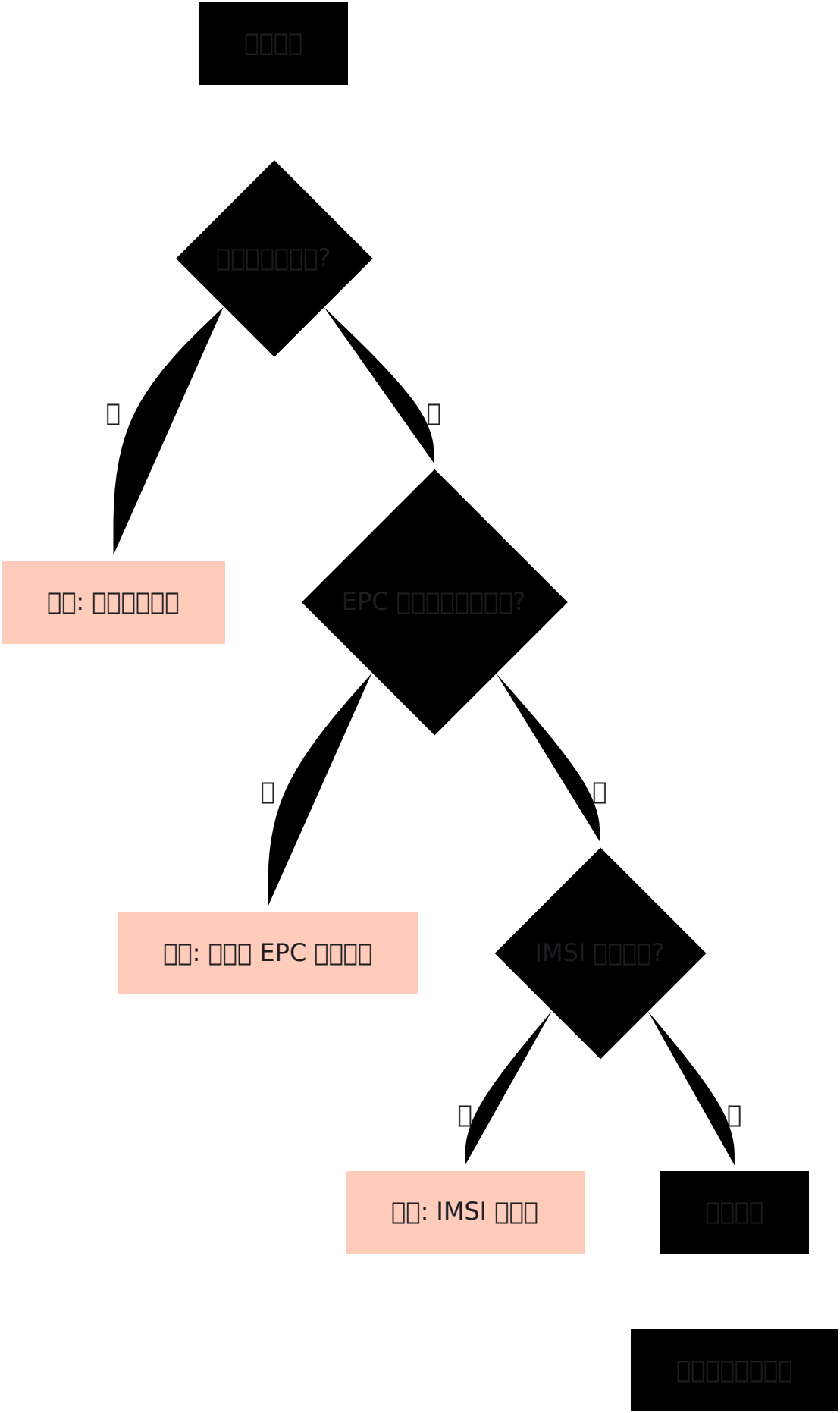
選擇:

- 32 位元組 - 32 位元組
- MSISDN 32 位元組 - 32 位元組
- IP 32 位元組 - IP 32 位元組 APN

選擇:

```
curl -k -X POST https://hss.example.com:8443/api/subscriber \
-H "Content-Type: application/json" \
-d '{
  "subscriber": {
    "imsi": "001001123456789",
    "key_set_id": "6ba7b810-9dad-41d1-80b4-00c04fd430c8",
    "epc_profile_id": "e58ed763-928c-4155-bee9-fdbaaadc15f3"
  }
}'
```

選擇:



接口

接口地址

请求: PUT /api/subscriber/:id

请求:

参数	类型	说明
id	string (UUID)	用户 ID

响应:

```
{
  "subscriber": {
    "enabled": false,
    "ims_enabled": false,
    "epc_profile_id": "f47ac10b-58cc-4372-a567-0e02b2c3d479",
    "custom_attributes": {
      "note": "测试"
    }
  }
}
```

响应:

- enabled - 是否启用
- ims_enabled - 是否启用 IMS
- sim_id - SIM 卡 ID
- key_set_id - 密钥集 ID
- epc_profile_id - EPC 配置文件 ID
- ims_profile_id - IMS 配置文件 ID
- roaming_profile_id - 漫游配置文件 ID

- `msisdns` - 電話番号 0000
- `static_ips` - 静的 APN 0 00 IP
- `custom_attributes` - カスタム属性

レスポンス:

- `imsi` - IMSI 番号

エラー:

- 400 - リクエストが無効

リクエスト:

```
curl -k -X PUT
https://hss.example.com:8443/api/subscriber/d290f1ee-6c54-4b01-
90e6-d701748f0851 \
-H "Content-Type: application/json" \
-d '{
  "subscriber": {
    "enabled": false
  }
}'
```

レスポンス:

- `subscriber`: {"enabled": false}
- `ims_enabled`: {"ims_enabled": false}
- `epc_profile_id`: {"epc_profile_id": "f47ac10b-58cc-4372-a567-0e02b2c3d479"} EPC プロファイル ID
- `roaming_profile_id`: {"roaming_profile_id": "1b671a64-40d5-491e-99b0-da01ff1f3341"} roaming プロファイル ID

リクエスト

レスポンス

DELETE /api/subscriber/:id

Request:

Field	Type	Description
id	string (UUID)	Subscriber ID

Response:

```
curl -k -X DELETE
https://hss.example.com:8443/api/subscriber/d290f1ee-6c54-4b01-
90e6-d701748f0851
```

Request: Delete subscriber PDN and IMSI

Request: Delete subscriber

- **PDN** - PDN address
- **SIM** - SIM card
- **IMSI** - IMSI number
- **MSISDNs** - MSISDN numbers

Response:

Response: CLR response MME

Request: POST /api/subscriber/cancel_location

Request:

```
{
  "imsi": "001001123456789"
}
```

Response:

필드	타입	길이	비고
imsi	string	15	14-15 자릿수 IMSI

요청:

```
curl -k -X POST
https://hss.example.com:8443/api/subscriber/cancel_location \
-H "Content-Type: application/json" \
-d '{"imsi": "001001123456789"}'
```

응답 (200 OK):

```
{
  "data": {
    "message": "Subscriber location cancelled",
    "imsi": "001001123456789",
    "destination_host": "mme01.operator.com",
    "destination_realm": "epc.operator.com"
  }
}
```

오류 (404 Not Found):

```
{
  "error": "Subscriber not found in MME"
}
```

참고:

- Subscriber MME에 subscriber_state.last_seen_mme 필드
- Cancellation-Type: subscription_withdrawal
- CLR-Flags: {s6a_indicator: 1, reattach_required: 1} UE가 재접속을 시도함
- last_seen_mme가 null이면 404
- IMSI와 MSISDNs를 사용하여 SIM을 식별

□□:

- □□□□: □□□□□□□□
- □□□□: □□□□□□□□□□
- □□□□: □□□□ MME □□□□□□
- □□: □□□□□□□□□□□□□□□□
- □□: □□□□□□□□□□

□ **IMSI** □□:

□□□ CLR □□□ MSISDN □□□□

1. □□ **MSISDN**□□□ **IMSI**:

```
// □□□ IMSI □ 001001123456789□MSISDNs □ ["+1234567890",  
"+9876543210"]  
POST /api/subscriber/cancel_location  
{  
  "imsi": "001001123456789"  
}  
  
// □□: □□□□ CLR□□□ MSISDN □□□□□□□□
```

2. □□ **IMSI**s□□□□□□:

```
// □□□□□□□□ MSISDN□□□□□ IMSIs□□□□□□□□  
// □□ A: IMSI 0010011111111111, MSISDN "+1234567890"  
// □□ B: IMSI 0010012222222222, MSISDN "+1234567890"  
  
POST /api/subscriber/cancel_location  
{  
  "imsi": "0010011111111111"  
}  
  
// □□: □□□ A □□□□□□ B □□□□
```

□□□□:

- □□ **IMSI**: CLR □□□ IMSI □□□□□□□ MSISDN
- □□: CLR □□□□□□□□□□□□□□ CLR □□□□□□□ MME □
- □□□ **MME** □□: □□ MME □□□□□ CLR □□□□□□□ HSS □□□
- □□: □□□ IMSI □□□□□□□□

요청:

- IMSI 목록
- IMSI
- S6a

Sh RepositoryData

Sh IMS-ODB Diameter Sh GET/PUT/DELETE
/api/subscriber/repository_data/:imsi

Sh RepositoryData REST API

MSISDN

MSISDN MSISDN

MSISDNs

GET /api/msisdn

```
curl -k https://hss.example.com:8443/api/msisdn
```

MSISDN

GET /api/msisdn/:id

curl:

```
curl -k https://hss.example.com:8443/api/msisdn/16fd2706-8baf-433b-82eb-8c7fada847da
```

MSISDN

MSISDN

Method: POST /api/msisdn

Request:

```
{
  "msisdn": {
    "msisdn": "14155551234"
  }
}
```

Response:

- 1-15 digits
- MSISDN
- E.164 MSISDN + extension

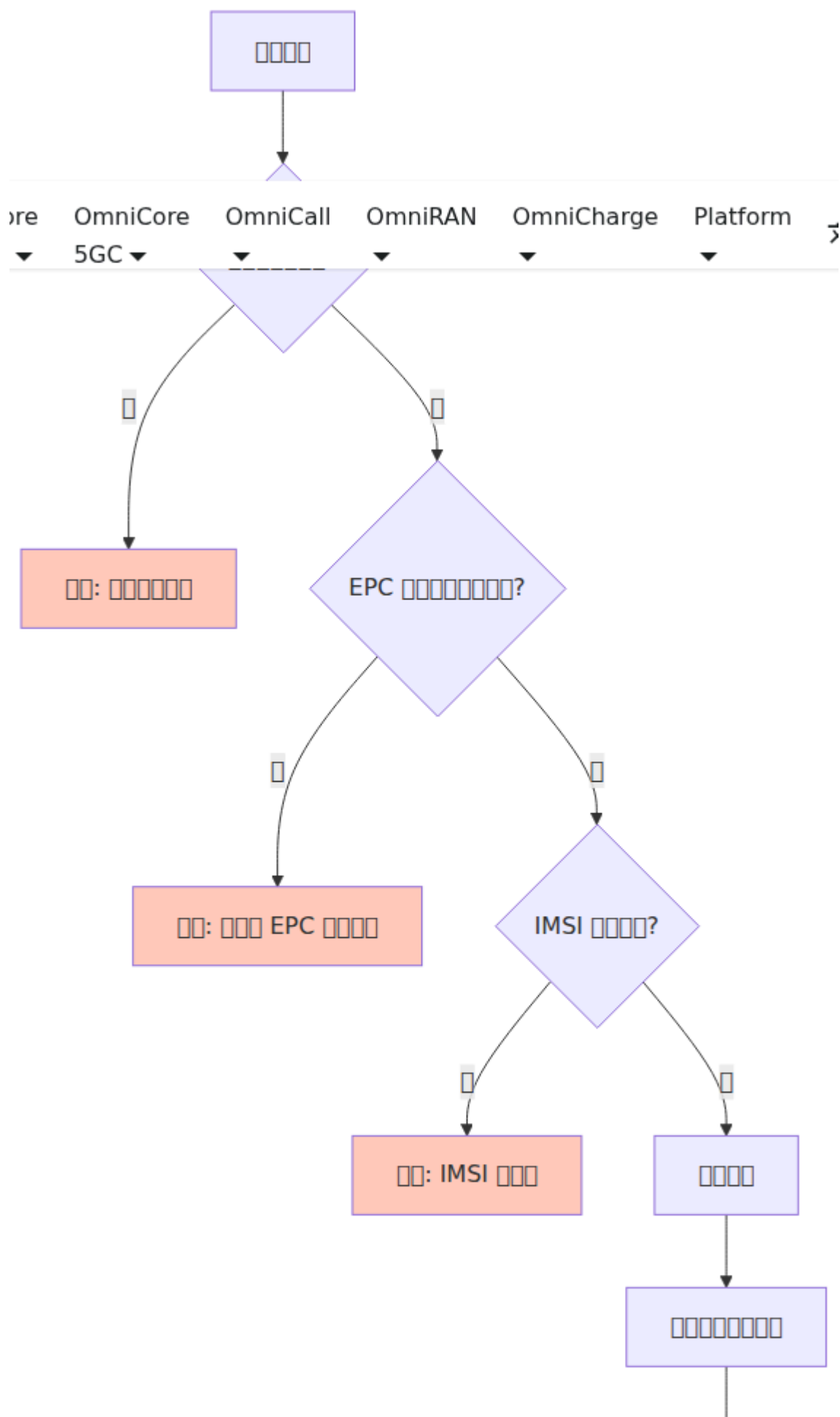
Example:

```
curl -k -X POST https://hss.example.com:8443/api/msisdn \
-H "Content-Type: application/json" \
-d '{
  "msisdn": {
    "msisdn": "14155551234"
  }
}'
```

MSISDN

[illegible]

MSISDN:





API 提供 MSISDN 及び IMSI に関する情報を返します

MSISDN

MSISDN 一覧を取得します

リクエスト: DELETE /api/msisdn/:id

レスポンス:

```
curl -k -X DELETE
https://hss.example.com:8443/api/msisdn/16fd2706-8baf-433b-82eb-
8c7fada847da
```

SIM

SIM に関する情報を返します。SIM に関する ICCID、PIN/PUK、OTA、SIM に関する情報を返します。

リクエスト:

- IMSI 提供 - SIM に関する情報を返します

SIMs

SIMs 一覧を取得します

リクエスト: GET /api/sim

レスポンス:

```
curl -k https://hss.example.com:8443/api/sim
```

GET SIM

Retrieve SIM details

Request: GET /api/sim/:id

Response:

```
curl -k https://hss.example.com:8443/api/sim/3f2504e0-4f89-41d3-9a0c-0305e82c3301
```

POST SIM

Create new SIM

Request: POST /api/sim

Response:

```
{
  "sim": {
    "iccid": "8991101200003204510",
    "sim_vendor": "Gemalto",
    "batch_name": "2025-Q1-Batch-01",
    "is_esim": false,
    "pin1": "1234",
    "pin2": "5678",
    "puk1": "12345678",
    "puk2": "87654321",
    "adm1": "admin-code-1",
    "kic": "0123456789ABCDEF0123456789ABCDEF",
    "kid": "FEDCBA9876543210FEDCBA9876543210"
  }
}
```

Response:

- iccid - 19-20 digit hexadecimal

Parameters:

- `sim_vendor` - Vendor
- `batch_name` - Batch
- `is_esim` - eSIM
- `pin1`, `pin2` - PIN
- `puk1`, `puk2` - PIN
- `adm1-adm10` - Admin
- `kic`, `kid` - OTA

Example:

```
curl -k -X POST https://hss.example.com:8443/api/sim \
-H "Content-Type: application/json" \
-d '{
  "sim": {
    "iccid": "8991101200003204510",
    "sim_vendor": "Gemalto"
  }
}'
```

Update SIM

Update SIM

Request: `PUT /api/sim/:id`

Example:

```
curl -k -X PUT https://hss.example.com:8443/api/sim/3f2504e0-4f89-
41d3-9a0c-0305e82c3301 \
-H "Content-Type: application/json" \
-d '{
  "sim": {
    "batch_name": "Batch"
  }
}'
```

🔑 SIM

🔑 SIM 🔑

🔑: DELETE /api/sim/:id

🔑: 🔑🔑🔑🔑🔑🔑🔑🔑🔑 SIM🔑

🔑🔑🔑🔑

🔑🔑🔑🔑🔑 Milenage 🔑🔑🔑🔑🔑🔑🔑🔑🔑Ki🔑OPC/OP🔑AMF🔑SQN🔑🔑🔑 🟡 🔑🔑🔑🔑🔑🔑🔑

🔑:

- 🟡 - 🔑🔑🔑🔑🔑🔑🔑🔑

❓❓❓🔑🔑🔑

🔑🔑🔑🔑🔑🔑🔑

🔑: GET /api/key_set

🔑🔑:

```
curl -k https://hss.example.com:8443/api/key_set
```

🔑🔑🔑🔑

🔑🔑🔑🔑🔑🔑

🔑: GET /api/key_set/:id

🔑🔑:

```
curl -k https://hss.example.com:8443/api/key_set/6ba7b810-9dad-41d1-80b4-00c04fd430c8
```

返回:

```
{
  "data": {
    "id": "6ba7b810-9dad-41d1-80b4-00c04fd430c8",
    "ki": "0123456789ABCDEF0123456789ABCDEF",
    "opc": "FEDCBA9876543210FEDCBA9876543210",
    "op": null,
    "amf": "8000",
    "sqn": 0,
    "authentication_algorithm": "milenege",
    "ota_counter": 0
  }
}
```

返回

返回

请求: POST /api/key_set

返回:

```
{
  "key_set": {
    "ki": "0123456789ABCDEF0123456789ABCDEF",
    "opc": "FEDCBA9876543210FEDCBA9876543210",
    "amf": "8000",
    "sqn": 0,
    "authentication_algorithm": "milenege"
  }
}
```

返回:

- `ki` - 128 bits 32 bytes
- `opc` & `op` OPC bits OP bits

Options

- `amf` - value "8000"
- `sqn` - value 0
- `ota_counter` - value 0

Options

- Authentication algorithm
- Ki, OPC, OP 32 bytes 128 bits
- AMF 4 bytes 16 bits

Options

```
curl -k -X POST https://hss.example.com:8443/api/key_set \
-H "Content-Type: application/json" \
-d '{
  "key_set": {
    "ki": "0123456789ABCDEF0123456789ABCDEF",
    "opc": "FEDCBA9876543210FEDCBA9876543210",
    "authentication_algorithm": "milenage"
  }
}'
```

Options Authentication algorithm API

Options

Options

PUT /api/key_set/:id

Options Authentication algorithm

Options Authentication algorithm

項目	単位	範囲	説明
name	文字列	文字	名前
ue_ambr_dl_kbps	文字列	Kbps	10000-1000000
ue_ambr_ul_kbps	文字列	Kbps	5000-500000
network_access_mode	文字列	文字	"network" or "internet"
tracking_area_update_interval_seconds	TAU間隔	秒	600秒
apn_profiles	APNプロファイルID	文字	[1, 2, 3]

例

```
curl -k -X POST https://hss.example.com:8443/api/epc/profile \
-H "Content-Type: application/json" \
-d '{
  "apn_profiles": [],
  "name": "100Mbps",
  "network_access_mode": "internet",
  "tracking_area_update_interval_seconds": 600,
  "ue_ambr_dl_kbps": 100000,
  "ue_ambr_ul_kbps": 50000
}'
```

例

- 100Mbps - 100Mbps
- 100Mbps - 100MbpsEPC

例EPC

PUT /api/epc/profile/:id

EPC

EPC

DELETE /api/epc/profile/:id

IMS

IMS IP IFC IMS

IMS

GET /api/ims/profile

IMS

POST /api/ims/profile

```
{
  "name": "VoLTE",
  "ifc_template": "<IMS-XML- ->"
}
```

- name -
- ifc_template - Liquid IFC XML

IFC

IFC Liquid

変数	説明	値
{{ imsi }}	IMSI	001001123456789
{{ msisdns }}	MSISDN	["14155551234", "14155555678"]
{{ mcc }}	MCC	001
{{ mnc }}	MNC	001

データ定義

IFCでLiquidテンプレートでjinja2でIMSIを生成する

- IMSIを生成するテンプレートで{{ imsi }}で{% for msisdn in msisdns %}
- APIでテンプレートでXMLを生成する
- IMSIをMAA/SAAでHSSに
◦ IMSを生成する
◦ テンプレートで生成する
 - {{ imsi }} → IMSI
 - {{ msisdns }} → MSISDN
 - {{ mcc }} → MCC
 - {{ mnc }} → MNC
◦ Cx/DiameterでXMLでS-CSCF

例

```
<!-- テンプレート -->
{{ imsi }}

<!-- テンプレート -->
{% for msisdn in msisdns %}
  <MSISDN>{{ msisdn }}</MSISDN>
{% endfor %}

<!-- テンプレート -->
{{ imsi }}@ims.mnc{{ mnc }}.mcc{{ mcc }}.3gppnetwork.org
```

IFC

```
<?xml version="1.0" encoding="UTF-8"?>
<IMSSubscription>
<PrivateID>{{ imsi }}@ims.mnc{{ mnc }}.mcc{{ mcc
}}.3gppnetwork.org</PrivateID>
<ServiceProfile>
{% for msisdn in msisdns %}
<PublicIdentity>
<Identity>sip:{{ msisdn }}@ims.mnc{{ mnc }}.mcc{{ mcc
}}.3gppnetwork.org</Identity>
<Extension>
<IdentityType>0</IdentityType>
</Extension>
</PublicIdentity>
<PublicIdentity>
<Identity>tel:{{ msisdn }}</Identity>
<Extension>
<IdentityType>0</IdentityType>
</Extension>
</PublicIdentity>
{% endfor %}
<InitialFilterCriteria>
<Priority>10</Priority>
<TriggerPoint>
<ConditionTypeCNF>0</ConditionTypeCNF>
<SPT>
<ConditionNegated>0</ConditionNegated>
<Group>0</Group>
<Method>REGISTER</Method>
</SPT>
</TriggerPoint>
<ApplicationServer>
<ServerName>sip:as.ims.mnc{{ mnc }}.mcc{{ mcc
}}.3gppnetwork.org</ServerName>
<DefaultHandling>0</DefaultHandling>
</ApplicationServer>
</InitialFilterCriteria>
</ServiceProfile>
</IMSSubscription>
```

curl

```
curl -k -X POST https://hss.example.com:8443/api/ims/profile \
-H "Content-Type: application/json" \
-d '{
  "name": "default",
  "ifc_template": "<?xml version=\"1.0\" encoding=\"UTF-8\"?>
<IMSSubscription><ServiceProfile>...</ServiceProfile>
</IMSSubscription>"
}'
```

Python

```
import requests

response = requests.post(
    "https://hss.example.com:8443/api/ims/profile",
    json={
        "name": "default",
        "ifc_template": ifc_template_string
    },
    verify=False # 
)
```

201

```
{
  "status": "success",
  "response": {
    "id": "0f8fad5b-d9cb-469f-a165-70867728950e",
    "name": "default",
    "ifc_template": "<?xml version=\"1.0\" encoding=\"UTF-8\"?
>..."
  }
}
```


- API IFC XML
-
- name

5/5

- 000000 - IFC0000000000
- 0000 - IMS00000000
- 00IFC00 - 0000

APN

APN

1. **APN** - APN/IP
2. **APN QoS** - QoS
3. **APN** - QoS **EPC**

PCRF QoS APN

APN

```
GET /api/apn/identifier
```

□□**APN**□□□

```
POST /api/apn/identifier
```

□□□□

```
{
  "apn": "internet",
  "ip_version": "ipv4v6"
}
```

IP

- "ipv4" - IPv4
- "ipv6" - IPv6
- "ipv4v6" - IPv4v6
- "ipv4_or_ipv6" - IPv4 or IPv6

API APN QoS

GET /api/apn/qos_profile

API APN QoS

POST /api/apn/qos_profile

Request

```
{
  "name": "APN QoS",
  "allocation_retention_priority": 8,
  "apn_ambr_dl_kbps": 50000,
  "apn_ambr_ul_kbps": 25000,
  "pre_emption_capability": false,
  "pre_emption_vulnerability": true,
  "qci": 9
}
```

API APN

GET /api/apn/profile

API APN

POST /api/apn/profile

Request

```
{
  "apn_identifier_id": "4fa85f64-5717-4562-b3fc-2c963f66afa6",
  "apn_qos_profile_id": "8d8ac610-566d-4ef0-9c22-186b2a5ed793",
  "name": "APN"
}
```

Response

- apn_identifier_id - APN ID
- apn_qos_profile_id - APN QoS ID

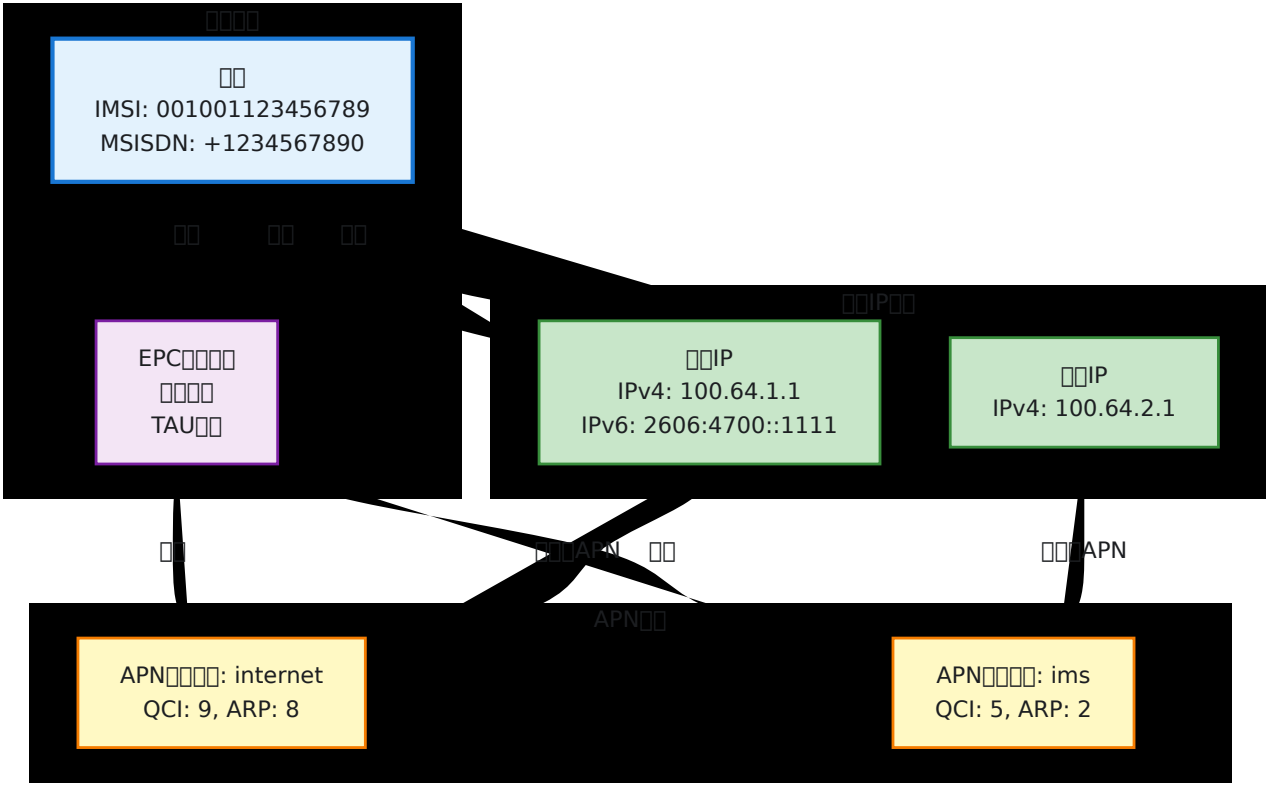
网络配置

- 网络配置 - 网络APN配置
- EPC配置 - APN配置EPC配置

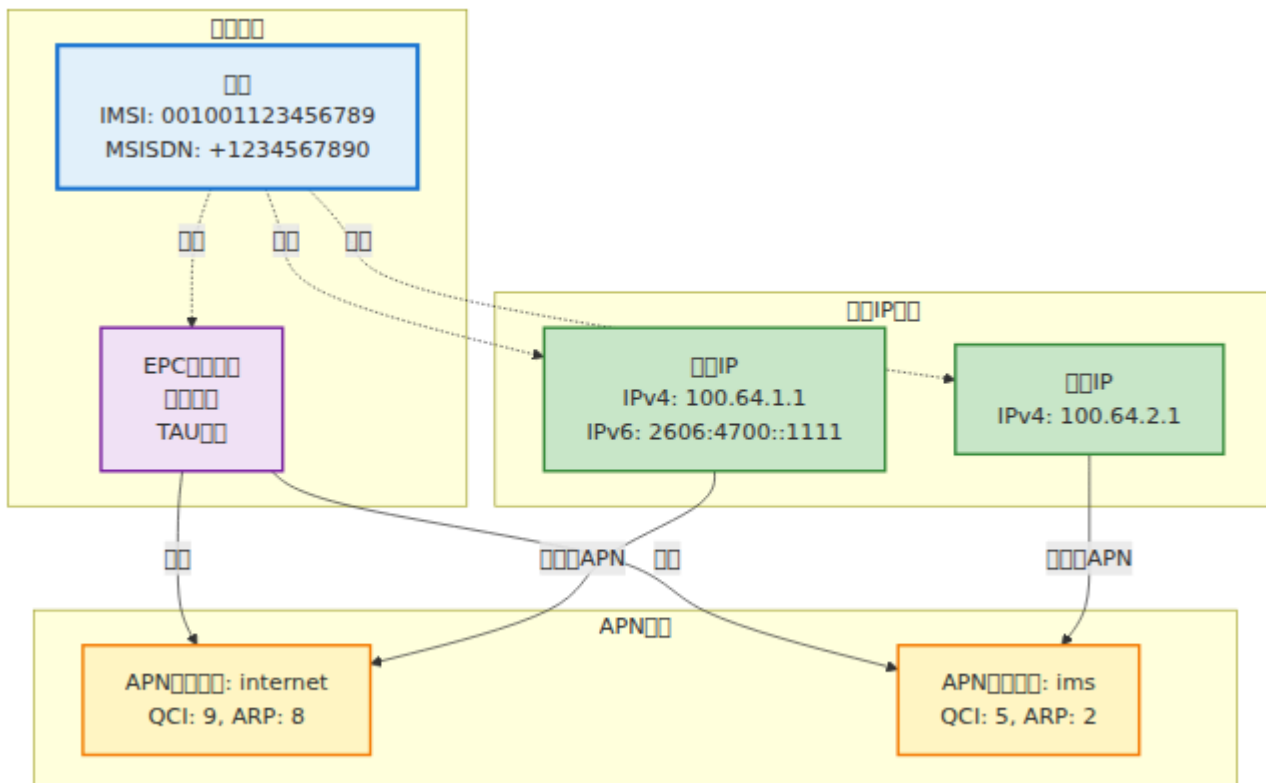
网络IP

网络IP配置网络APN配置网络APN配置IPv4/IPv6配置DHCP配置

网络



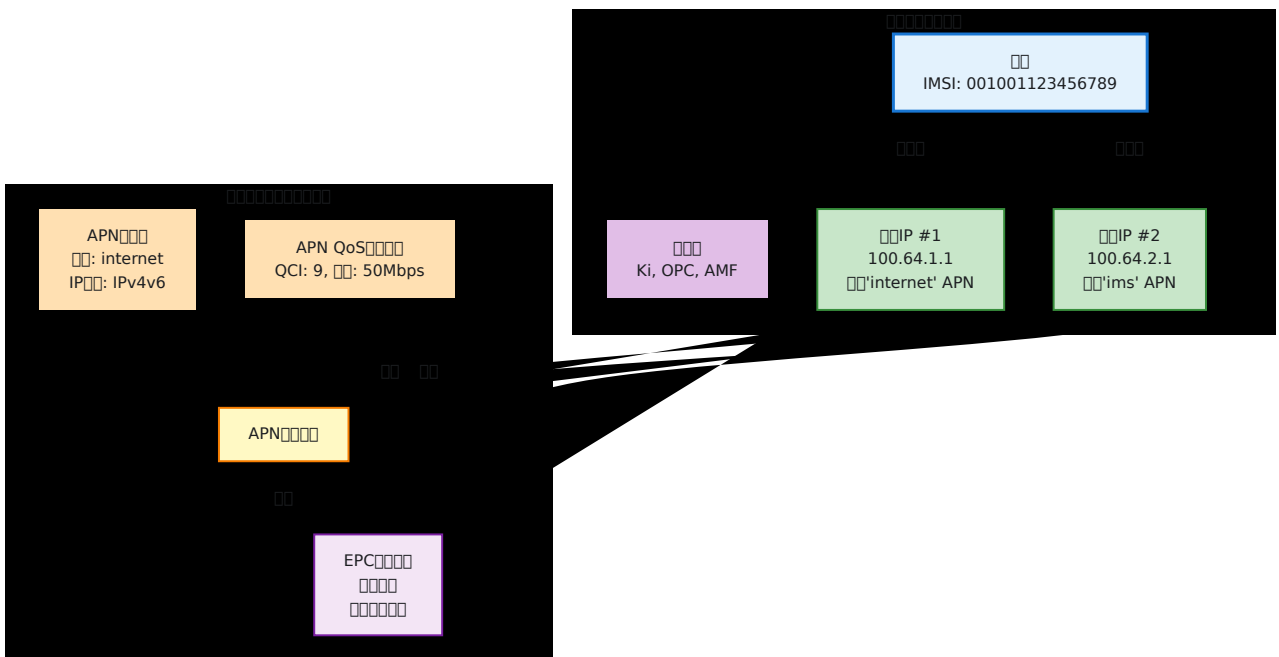
网络配置



□□□□□

1. **APN** 0, 1, 2...
2. **apn_identifier.apn** "internet" "ims"
3. **PDN** **apn_identifier.ip_version** ipv4=0 ipv6=1 ipv4v6=2
ipv4_or_ipv6=3
4. **QoS** **apn_qos_profile**
5. **AMBR** 1000 kbps → bps
6. **IP** + APN IP
 - **subscriber.static_ips** → **apn_profile_id** → IP
 - **apn_identifier.ip_version**
7. **VPLMN**-0 -   IP

□ □ □ □ □ □ □



□□□□□

- **APN** IP **APN**
- **APN** IP APN IP
- **IPv4** **IPv6** IP IPv4 IPv6
- IP IP IP
 - IPv4 IPv6 APN
 - IP

◦ ipv4_static_ip ipv6_static_ip

- IP

- IoT IP
- IP
- IP
- IP
- IP

IP

IP

GET /api/epc/static_ip

```
curl -k https://hss.example.com:8443/api/epc/static_ip
```

```
{
  "data": [
    {
      "id": "c56a4180-65aa-42ec-a945-5fd21dec0538",
      "apn_profile_id": "a1b2c3d4-e5f6-4a3b-8c2d-1e0f9a8b7c6d",
      "ipv4_static_ip": "100.64.1.1",
      "ipv6_static_ip": "2606:4700:4700::1111",
      "apn_profile": {
        "id": "a1b2c3d4-e5f6-4a3b-8c2d-1e0f9a8b7c6d",
        "name": "测试APN",
        "apn_identifier": {
          "apn": "internet",
          "ip_version": "ipv4v6"
        }
      },
      "inserted_at": "2025-11-15T10:30:00Z",
      "updated_at": "2025-11-15T10:30:00Z"
    }
  ]
}
```

测试IP

测试IP接口

GET /api/epc/static_ip/:id

返回

参数	类型	说明
id	string (UUID)	IP ID

示例

```
curl -k https://hss.example.com:8443/api/epc/static_ip/c56a4180-65aa-42ec-a945-5fd21dec0538
```

Static IP

APN Static IP

POST /api/epc/static_ip

Request

```
{
  "static_ip": {
    "apn_profile_id": "a1b2c3d4-e5f6-4a3b-8c2d-1e0f9a8b7c6d",
    "ipv4_static_ip": "100.64.1.1",
    "ipv6_static_ip": "2606:4700:4700::1111"
  }
}
```

Response

- apn_profile_id - APN ID
- ipv4_static_ip ipv6_static_ip

Fields

- ipv4_static_ip - IPv4 address
- ipv6_static_ip - IPv6 address

IP Address

- IPv4 address 100.64.1.1
- IPv6 address 2606:4700:4700::1111
- IPv4 or IPv6 IP address
 - IP address
 - IP address APN
 - IP address

Response

項目	IPv4	IPv6	値
IPv4	✓	-	<code>{"ipv4_static_ip": "100.64.1.1"}</code>
IPv6	-	✓	<code>{"ipv6_static_ip": "2606:4700:4700::1111"}</code>
	✓	✓	

IPv4 IP

```
curl -k -X POST https://hss.example.com:8443/api/epc/static_ip \
-H "Content-Type: application/json" \
-d '{
  "static_ip": {
    "apn_profile_id": "a1b2c3d4-e5f6-4a3b-8c2d-1e0f9a8b7c6d",
    "ipv4_static_ip": "100.64.1.1"
  }
}'
```

IPv6 IP

```
curl -k -X POST https://hss.example.com:8443/api/epc/static_ip \
-H "Content-Type: application/json" \
-d '{
  "static_ip": {
    "apn_profile_id": "b2c3d4e5-f6a7-4b1c-9d2e-3f4a5b6c7d8e",
    "ipv6_static_ip": "2606:4700:4700::1111"
  }
}'
```

IP

```
curl -k -X POST https://hss.example.com:8443/api/epc/static_ip \
-H "Content-Type: application/json" \
-d '{
  "static_ip": {
    "apn_profile_id": "a1b2c3d4-e5f6-4a3b-8c2d-1e0f9a8b7c6d",
    "ipv4_static_ip": "100.64.1.1",
    "ipv6_static_ip": "2606:4700:4700::1111"
  }
}'
```

レスポンス (201 OK)

```
{
  "data": {
    "id": "c56a4180-65aa-42ec-a945-5fd21dec0538",
    "apn_profile_id": "a1b2c3d4-e5f6-4a3b-8c2d-1e0f9a8b7c6d",
    "ipv4_static_ip": "100.64.1.1",
    "ipv6_static_ip": "2606:4700:4700::1111",
    "inserted_at": "2025-11-15T10:30:00Z",
    "updated_at": "2025-11-15T10:30:00Z"
  }
}
```

注意事項

- IP アドレス - 一意に決まります
- APN アドレス - 1つのAPNのみ

IP アドレス

一意に決まる IP アドレス

PUT /api/epc/static_ip/:id

レスポンス

項目	型別	説明
id	string (UUID)	静的 IP の ID

例

```
{
  "static_ip": {
    "ipv4_static_ip": "100.64.1.2",
    "ipv6_static_ip": "2606:4700:4700::1112"
  }
}
```

フィールド

- ipv4_static_ip - IPv4 の静的 IP
- ipv6_static_ip - IPv6 の静的 IP
- apn_profile_id - APN の ID

レスポンス

- id - 静的 IP の ID

静的 IP を作成する API のリクエストの例

例

```
curl -k -X PUT
https://hss.example.com:8443/api/epc/static_ip/c56a4180-65aa-42ec-
a945-5fd21dec0538 \
-H "Content-Type: application/json" \
-d '{
  "static_ip": {
    "ipv4_static_ip": "100.64.1.2"
  }
}'
```


刪除 IP

刪除 IP 詳細

DELETE /api/epc/static_ip/:id

刪除 IP

欄位	資料型態	說明
id	string (UUID)	IP 的 ID

刪除 IP

```
curl -k -X DELETE https://hss.example.com:8443/api/epc/static_ip/c56a4180-65aa-42ec-a945-5fd21dec0538
```

刪除 IP 後

- 刪除 IP 後
- 刪除 APN 後 APN 狀態會變成已刪除
- 刪除 IP 後 IP 狀態會變成已刪除
- 刪除 IP 後 IP 狀態會變成已刪除

刪除 IP 後 IP 狀態會變成已刪除 PDN 狀態會變成已刪除 IP 狀態會變成已刪除

刪除 IP 詳細

刪除 IP 詳細 IP 狀態 刪除 IP 詳細

刪除 IP

- 刪除 IP 狀態 刪除 IP
- 刪除 static_ips 狀態

刪除 IP 詳細

```
# 1 创建 "internet" APN 下的 IP
STATIC_IP_ID=$(curl -k -X POST
https://hss.example.com:8443/api/epc/static_ip \
-H "Content-Type: application/json" \
-d '{
  "static_ip": {
    "apn_profile_id": "a1b2c3d4-e5f6-4a3b-8c2d-1e0f9a8b7c6d",
    "ipv4_static_ip": "100.64.1.1",
    "ipv6_static_ip": "2606:4700:4700::1111"
  }
}' | jq -r '.data.id')

# 2 创建静态 IP 配置
curl -k -X POST https://hss.example.com:8443/api/subscriber \
-H "Content-Type: application/json" \
-d "{
  \"subscriber\": {
    \"imsi\": \"001001123456789\",
    \"key_set_id\": \"6ba7b810-9dad-41d1-80b4-00c04fd430c8\",
    \"epc_profile_id\": \"e58ed763-928c-4155-bee9-fdbaaadc15f3\",
    \"static_ips\": [\$STATIC_IP_ID]
  }
}"
```

创建静态 IP

```
curl -k -X PUT
https://hss.example.com:8443/api/subscriber/d290f1ee-6c54-4b01-90e6-d701748f0851 \
-H "Content-Type: application/json" \
-d '{
  "subscriber": {
    "static_ips": ["c56a4180-65aa-42ec-a945-5fd21dec0538",
"5a8f9c3e-2d4b-4e6f-8a1b-3c5d7e9f0a2b"]
  }
}'
```

创建 IP 下的 APN

创建静态 IP 下的 APN

```
# [] "internet" APN [] IP
INTERNET_IP=$(curl -k -X POST
https://hss.example.com:8443/api/epc/static_ip \
-H "Content-Type: application/json" \
-d '{
  "static_ip": {
    "apn_profile_id": "a1b2c3d4-e5f6-4a3b-8c2d-1e0f9a8b7c6d",
    "ipv4_static_ip": "100.64.1.1"
  }
}' | jq -r '.data.id')

# [] "ims" APN [] IP
IMS_IP=$(curl -k -X POST
https://hss.example.com:8443/api/epc/static_ip \
-H "Content-Type: application/json" \
-d '{
  "static_ip": {
    "apn_profile_id": "b2c3d4e5-f6a7-4b1c-9d2e-3f4a5b6c7d8e",
    "ipv4_static_ip": "100.64.2.1"
  }
}' | jq -r '.data.id')

# [] IP []
curl -k -X POST https://hss.example.com:8443/api/subscriber \
-H "Content-Type: application/json" \
-d "{
  \"subscriber\": {
    \"imsi\": \"001001123456789\",
    \"key_set_id\": \"6ba7b810-9dad-41d1-80b4-00c04fd430c8\",
    \"epc_profile_id\": \"e58ed763-928c-4155-bee9-
fdbaaadc15f3\",
    \"static_ips\": [$INTERNET_IP, $IMS_IP]
  }
}"
```

□□□□

- ✓ □□□□ APN □□□□ IP
- ✗ □□□□ APN □□□□ IP

□□□□ - □□ **APN**

```
# 000000 IP 000000 APN000000
curl -k -X POST https://hss.example.com:8443/api/subscriber \
-H "Content-Type: application/json" \
-d '{
  "subscriber": {
    "imsi": "001001123456789",
    "static_ips": ["c56a4180-65aa-42ec-a945-5fd21dec0538",
"5a8f9c3e-2d4b-4e6f-8a1b-3c5d7e9f0a2b"]
  }
}'

# 000000
{
  "errors": {
    "static_ips": [
      "00000000 APN 0000 IP 0000000000000000 internet 0000 IP
100.64.1.100000 internet 00 100.64.1.2"
    ]
  }
}
```

000000

- 000000 - 000000
- 000000 - 00000000
- 0000 IP 0000 - 000000

000000

0000000000000000000000000000 IMS 0000000000 000000 MCC/MNC 00000000

0000000000

000 GET /api/roaming/profile

POST /api/roaming/profile

POST /api/roaming/profile

Request

```
{
  "roaming_profile": {
    "name": "Roaming Profile",
    "data_action_if_no_rules_match": "deny",
    "ims_action_if_no_rules_match": "deny",
    "roaming_rules": []
  }
}
```

Response

- "allow" - true
- "deny" - false

Request

- data_action_if_no_rules_match - Data action if no rules match
- ims_action_if_no_rules_match - IMS action if no rules match

GET /api/roaming/rule

GET /api/roaming/rule

POST /api/roaming/rule

POST /api/roaming/rule

Request

```
{
  "roaming_rule": {
    "name": "AT&T",
    "mcc": "310",
    "mnc": "410",
    "data_action": "allow",
    "ims_action": "allow"
  }
}
```

필드

- `mcc` - 3자리
- `mnc` - 2-3자리
- `data_action` - "allow" 또는 "deny"
- `ims_action` - "allow" 또는 "deny" IMS/

참고

- -
- -

EIR

OmniHSS S13 EIR IMEI

S13 IMEI EIR

EIR

GET /api/eir/rule

EIR

POST /api/eir/rule

設定

```
{
  "eir_rule": {
    "name": "iPhone 6",
    "imei_regex": "^35[0-9]{6}0[0-9]{7}$",
    "action": 1
  }
}
```

設定

- name - 名前
- imei_regex - IMEI の正規表現
- action - 0: 警告 1: 拒否 2: 削除

設定

- 0 - 警告
- 1 - 拒否
- 2 - 削除

設定

- IMEI の正規表現
- TAC の正規表現
- 拒否理由

設定

- S13 EIR 設定
- OmniHSS EIR 設定

設定

設定

- **API** - API
- **API** - API
- **API** - API

← API | API →

API 入門

← API 入門

API

API 入門

GET /api/status

API

```
curl -k https://hss.example.com:8443/api/status
```

API

```
{  
  "status": "ok"  
}
```

API 入門

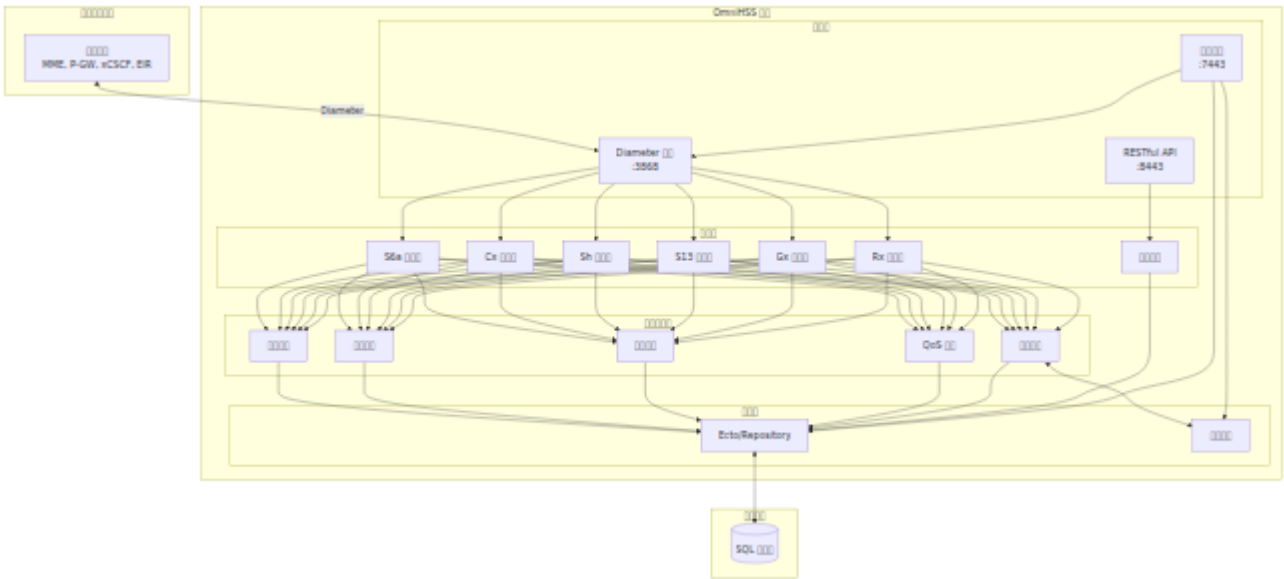
← API 入門

OmniHSS

←

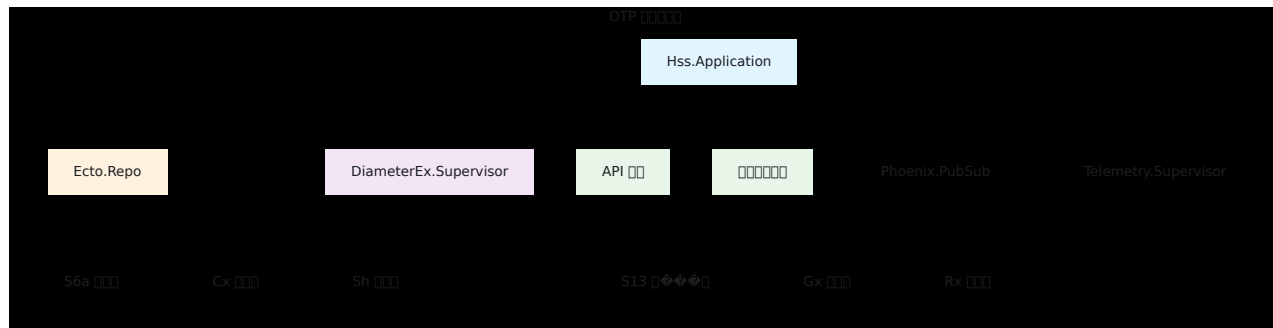
-
-
- Diameter
-
-
-
-

OmniHSS is an Elixir / Erlang/OTP based distributed system architecture for 4G/LTE network core.



□□□□

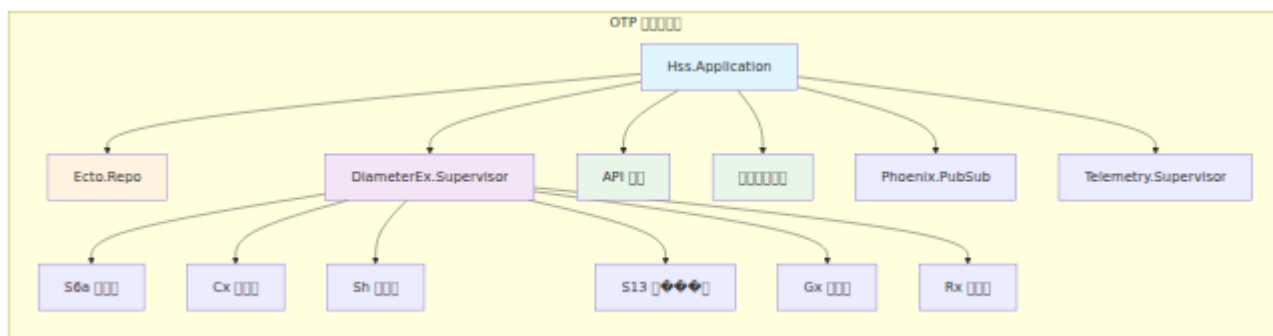
□□□□



Diameter □□□□

□□ Diameter □□□S6a, Cx, Sh, S13, Gx, Rx□□□□□□ DiameterEx □□□□□□□□□□

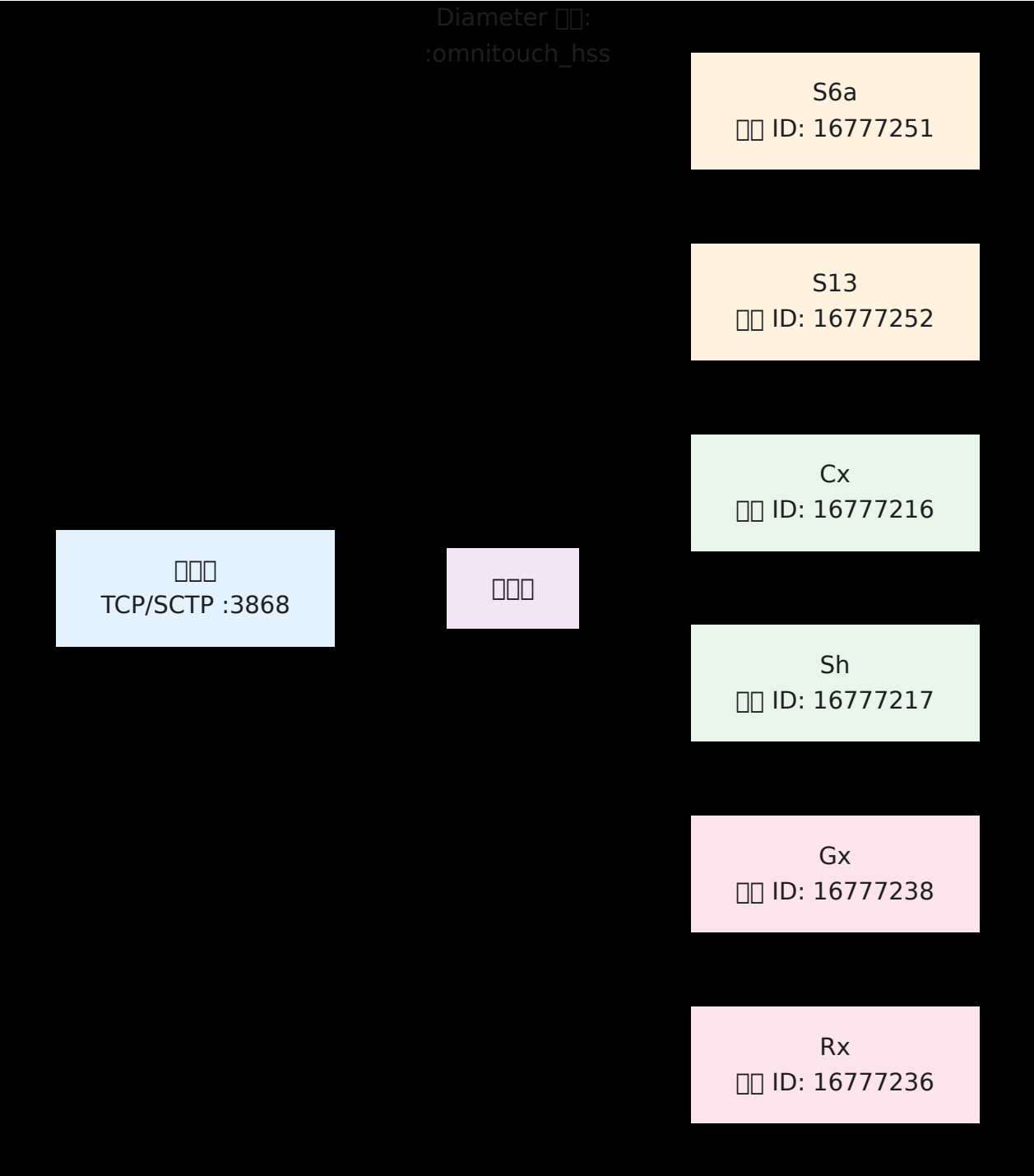
1. □ **DiameterEx** □□ - □□□□□ Diameter □□ ID
2. □□□□ - □□ AVP□□□□□□□□
3. □□□□□□ - □□□□□□□□□□□□
4. □□□□ - □□ AVP □□ Diameter □□□□
5. □□□□ - □□□□□□ Diameter □□□□



Diameter

Diameter

OmniHSS Diameter



□□□□□□



□□□□□□

Configured

□□□□

Connecting

□□□□

Connected

□□□□

□□□□□□

□□□□□□

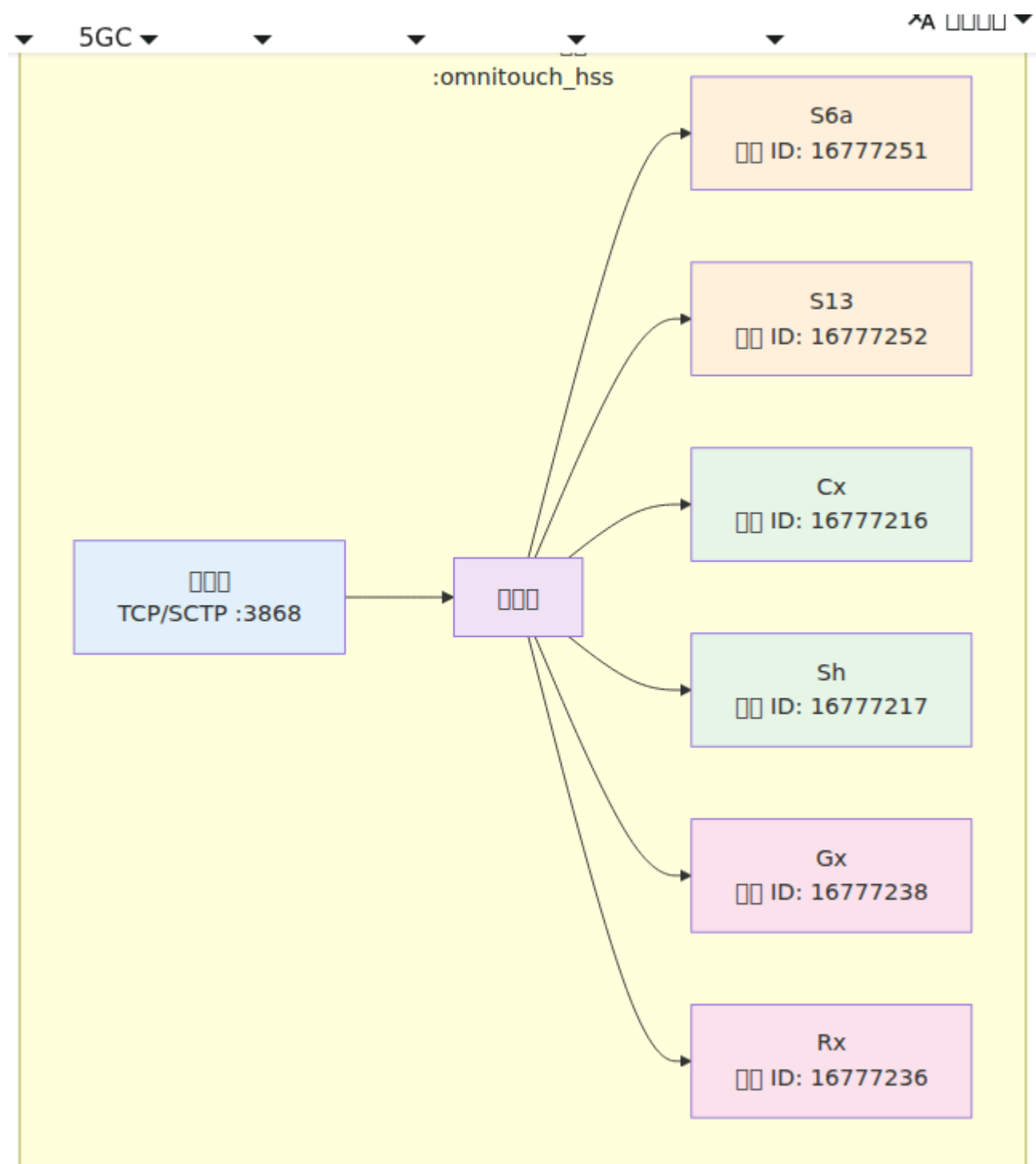
□□□□

□□□□ Diameter □□

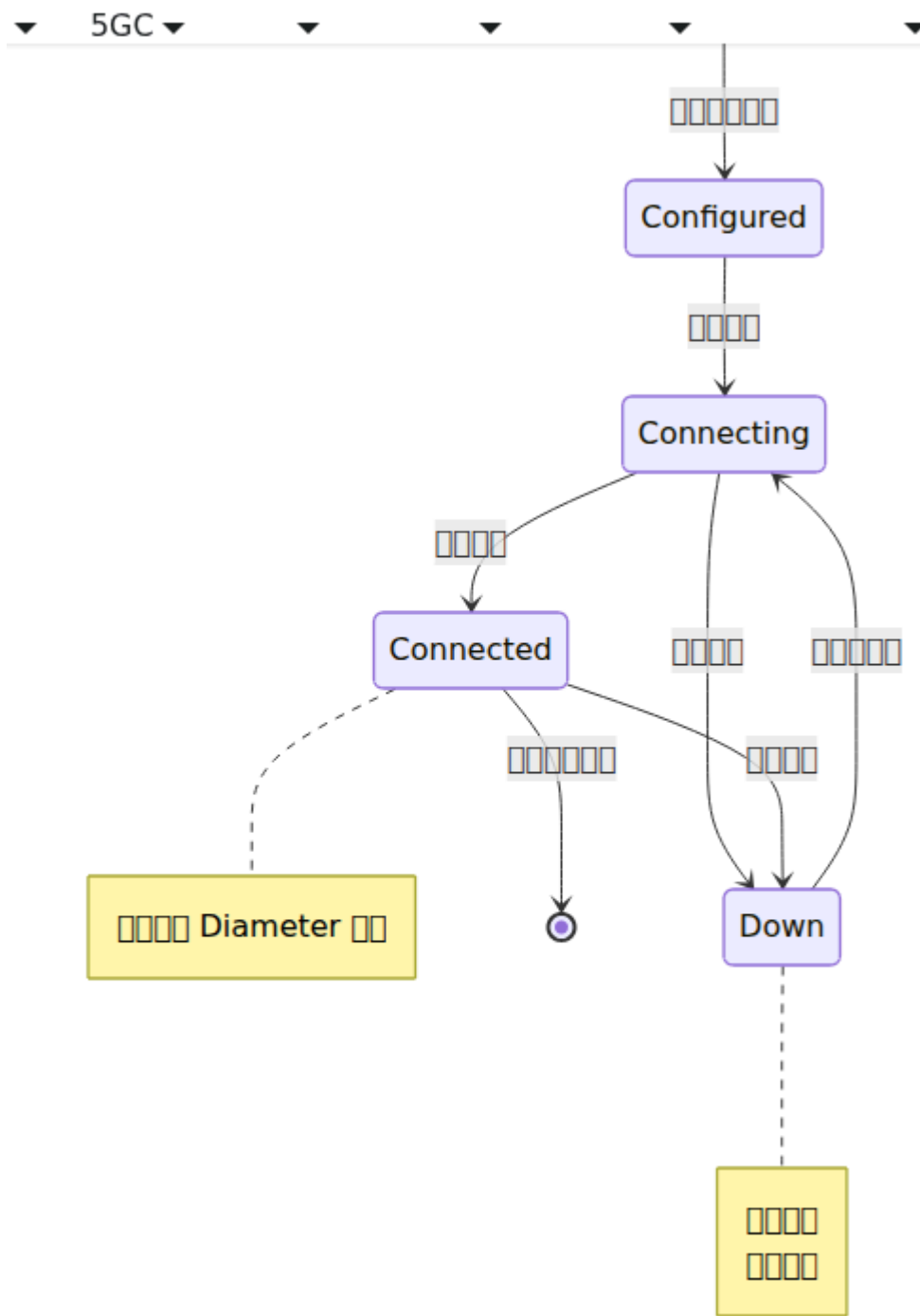


Down

□□□□
□□□□

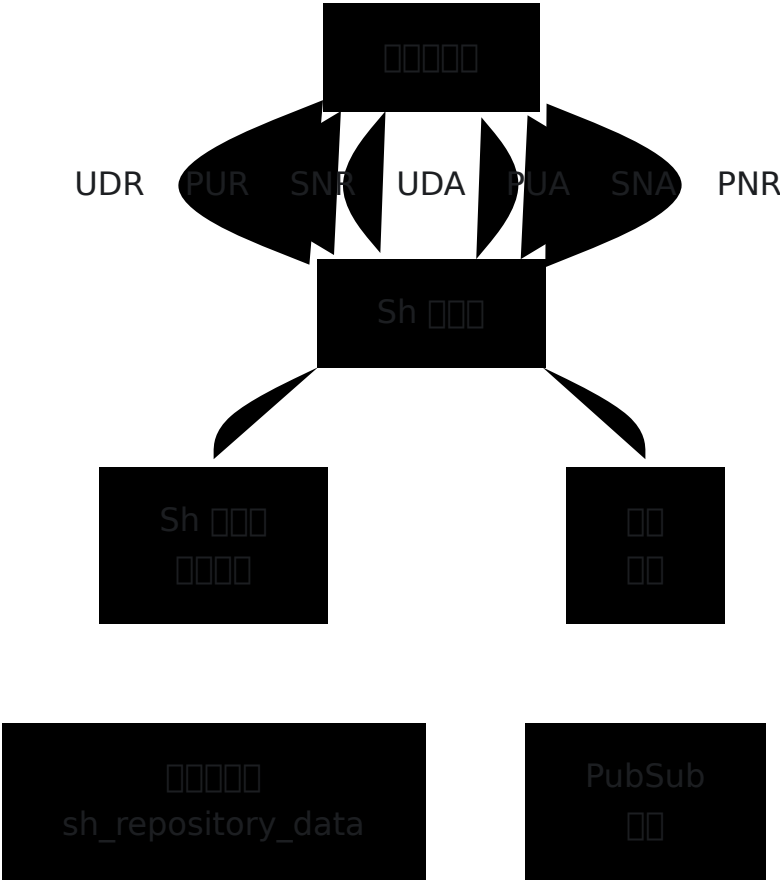
Diameter 

IMS 000000??



Sh (IMS)

IMS

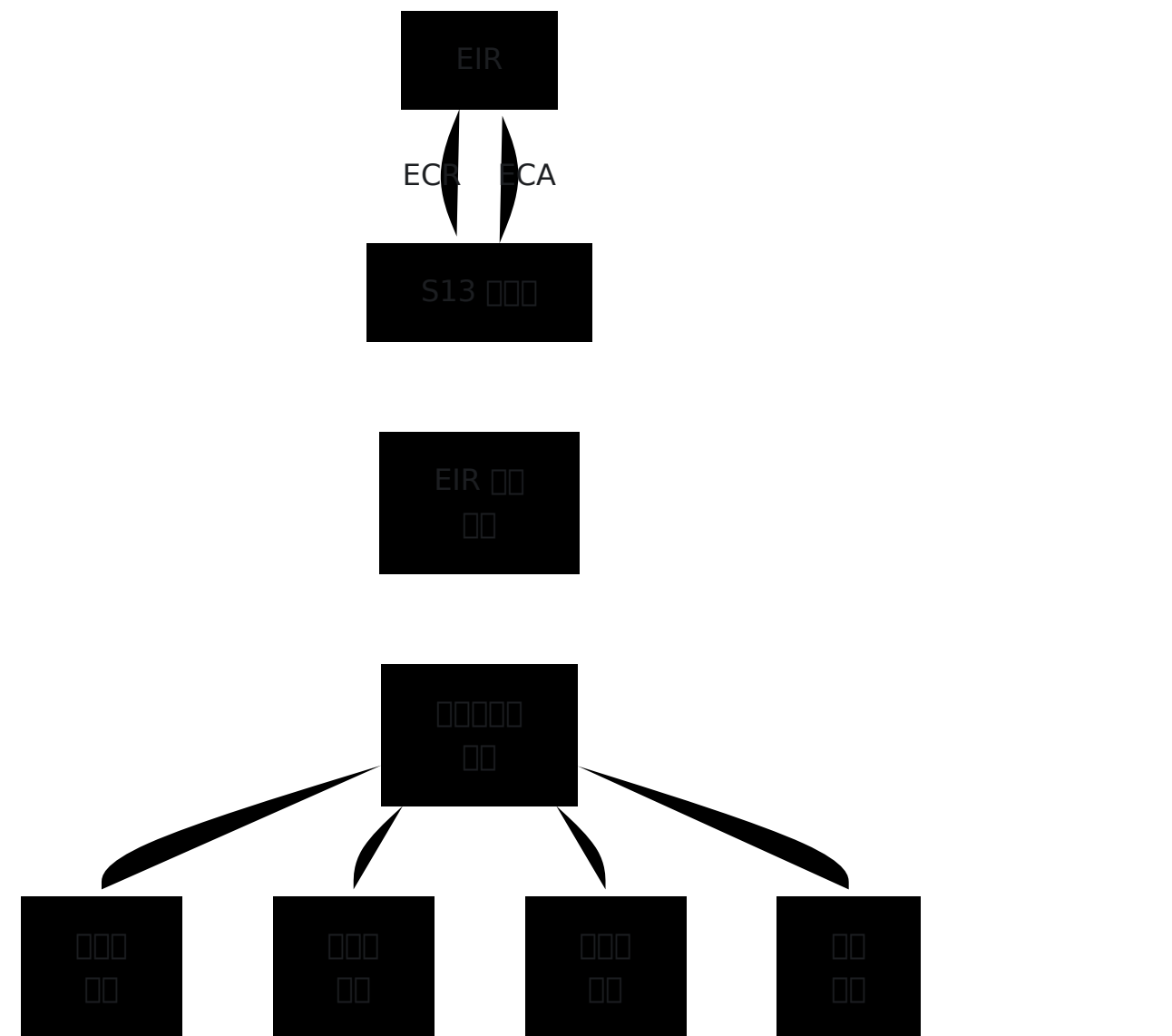


Gx ()

PCRF

IMS VoLTE PCRF

IMEI EIR



□□□

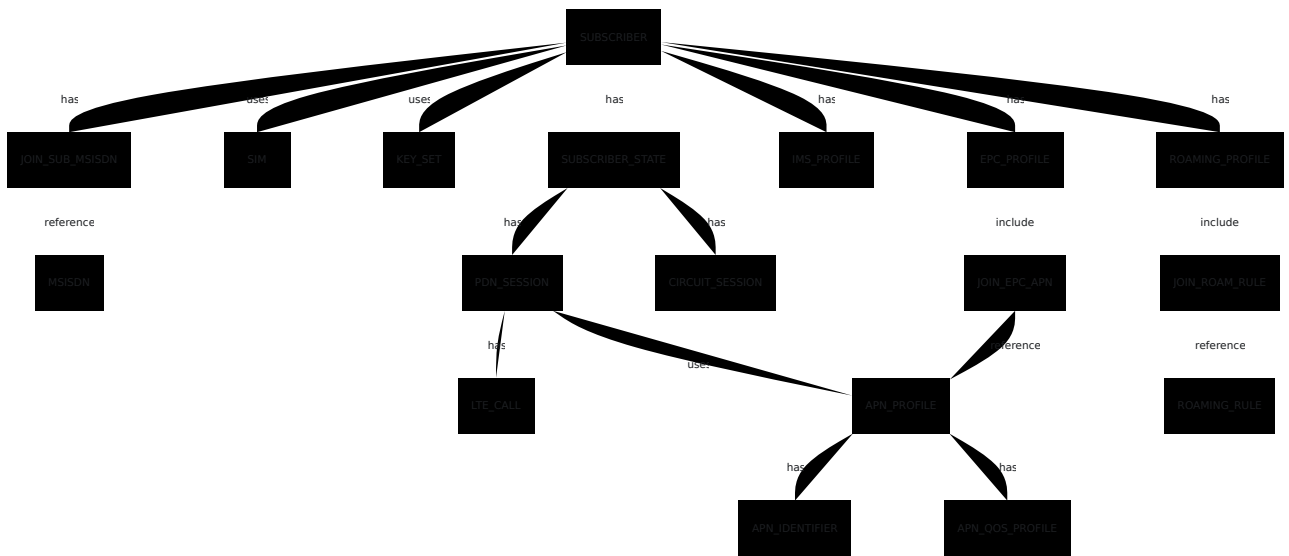
□□□□□

OmniHSS 的 **PostgreSQL** 数据库使用 **Ecto** 数据库框架使用 **UUID** 数据库主键
数据库主键使用数据库主键

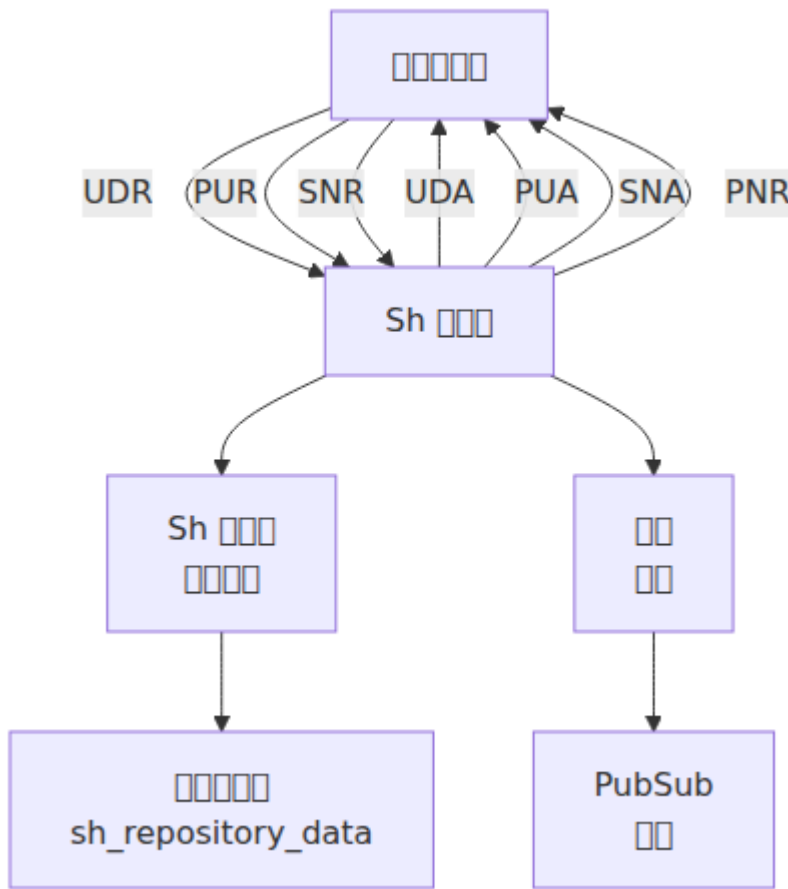
数据库使用 OmniHSS 的 PostgreSQL 数据库主键使用数据库主键/数据库主键使用数据库主键
WAL 数据库主键

数据库主键使用数据库主键 数据库主键

□□□□□□□□

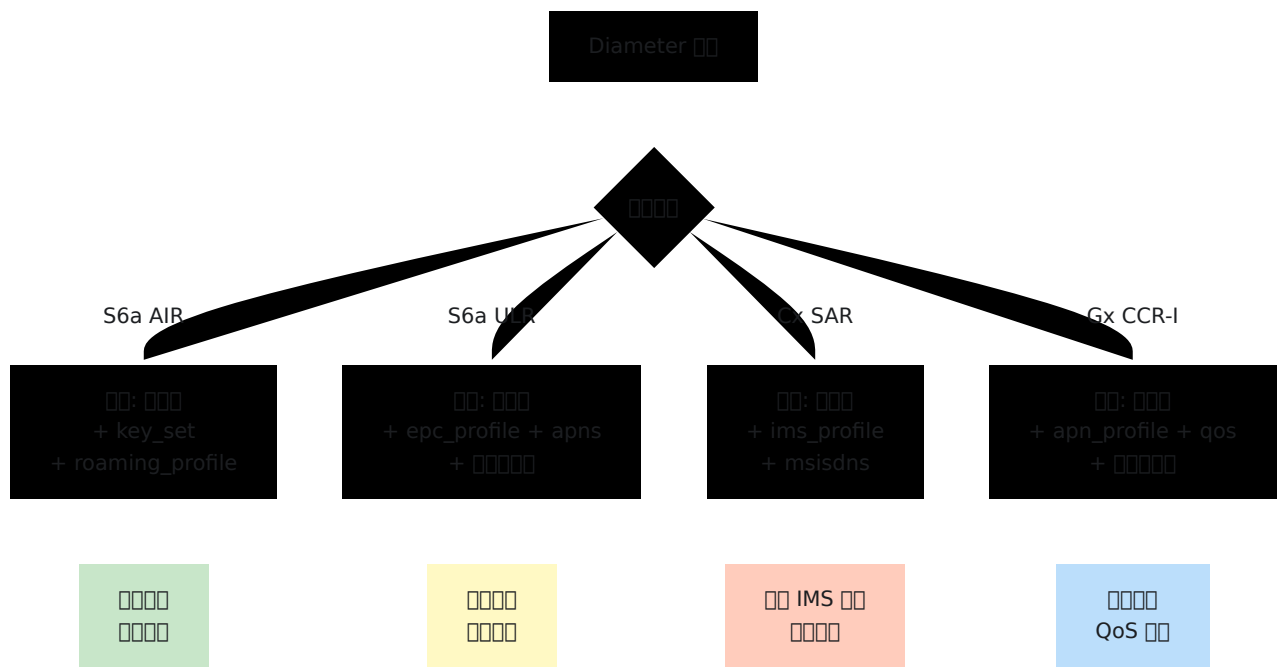


Ecto □□□□



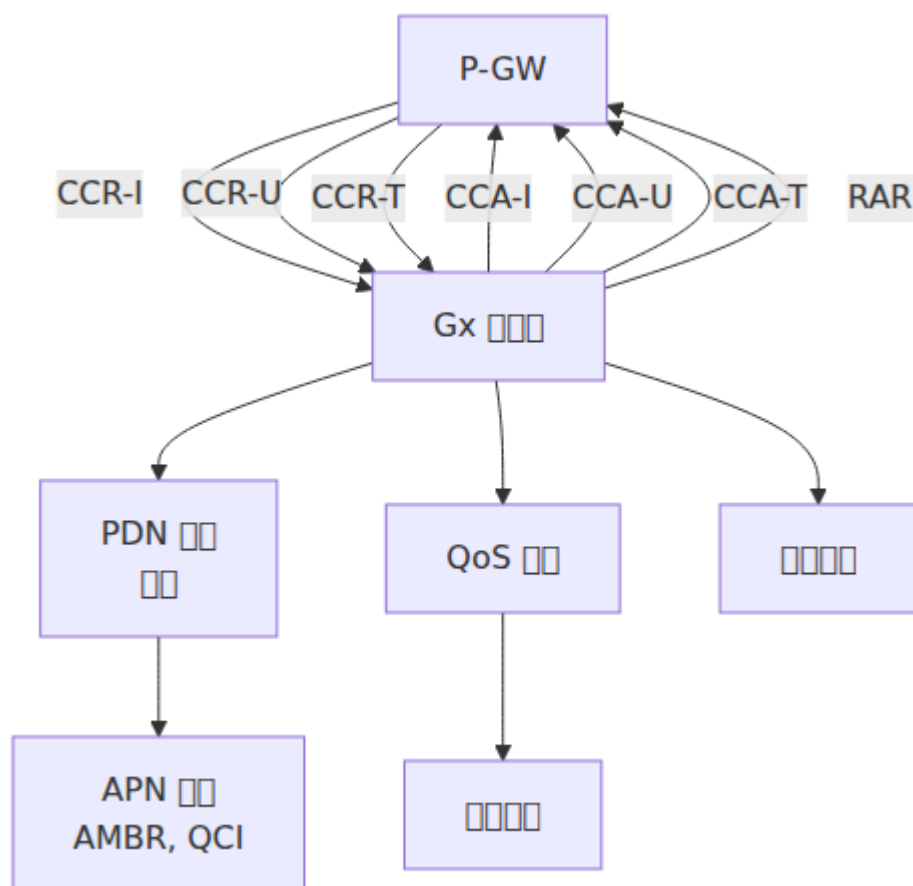
□□□□□□

□□ Diameter □□□□□□□□□□□□□□□□□□

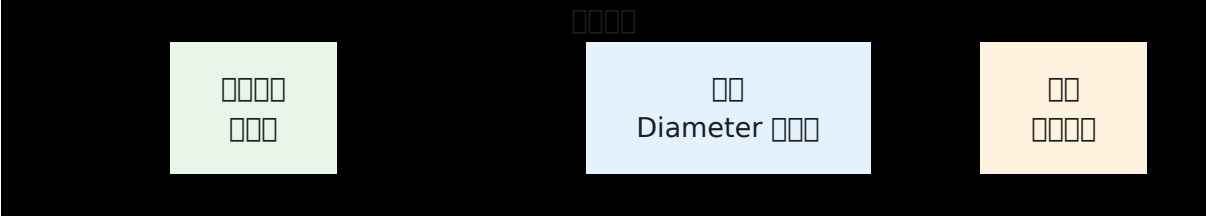
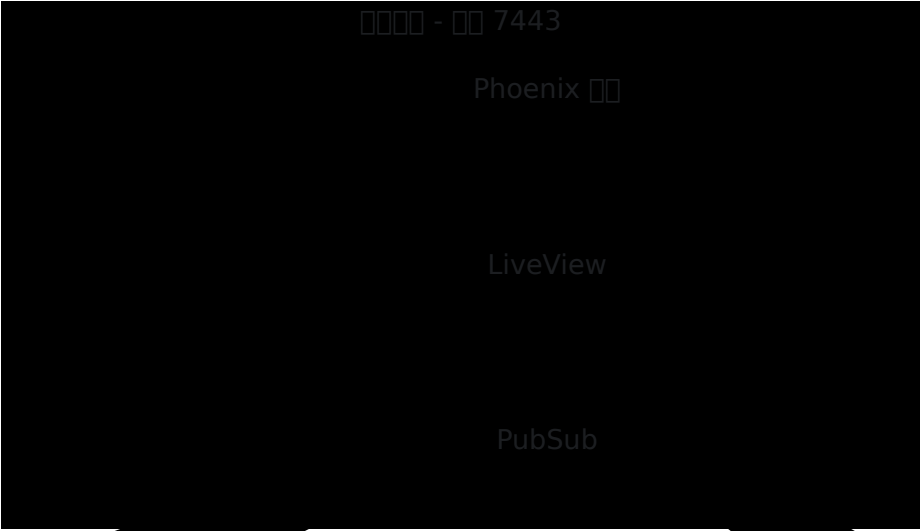


□□□□

API □□



Phoenix 7443

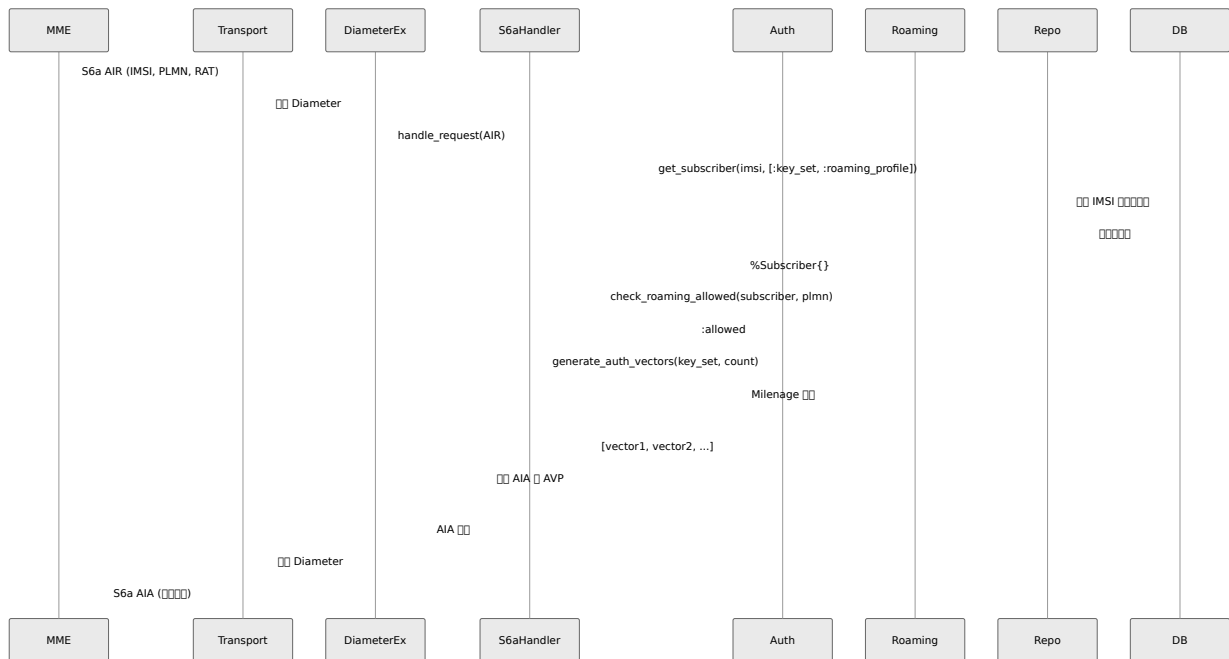




-

Sequence Diagram

Sequence Diagram



Sequence Diagram

1. Sequence

- Erlang/OTP sequence diagram
- Diameter sequence diagram
- Sequence diagram

2. Sequence

- Diameter sequence diagram
- Sequence diagram
- Sequence diagram

3. 鋼管

- 鋼管 Diameter 鋼管規格
- 鋼管規格表
- 鋼管規格

4. 鋼管

- 鋼管規格表
- 鋼管規格
- 鋼管規格

5. 鋼管

- 鋼管規格
- 鋼管規格
- Diameter 鋼管規格
- 鋼管規格

OmniHSS 架构图

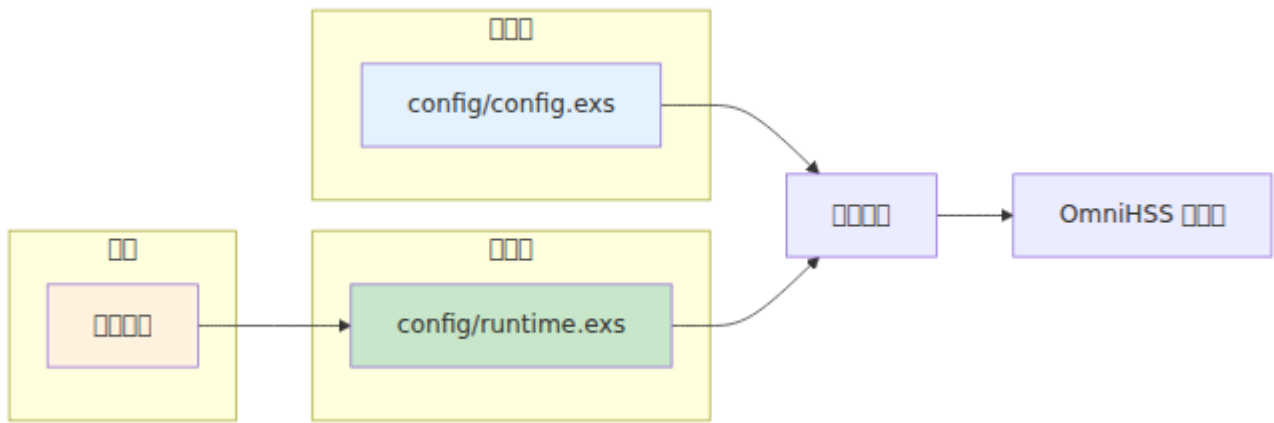
← 架构图

图例

- 数据库
- 网络元素
- 接口
- 协议
- Diameter 接口
- 接口
 - 与 PLMN 接口
 - HSS 接口
- IMS 接口
- EIR 接口
- API 接口
- 接口

架构图

OmniHSS 架构图



config/config.exs (config)

config/config.exs is a file that defines the configuration for the application.

- config/config.exs
- API config
- config

config/runtime.exs (runtime)

config/runtime.exs is a file that defines the configuration for the application at runtime.

- config/runtime.exs
- Diameter config
- PLMN config
- IMS S-CSCF config
- config

config/runtime.exs (runtime)

config/runtime.exs is a file that defines the configuration for the application at runtime.

```
# config/runtime.exs

config :license_client,
  # 许可证 API 端点地址
  license_server_api_urls:
  ["https://license.example.com:8443/api"],

  # 许可证持有者
  licensee: "许可证",

  # 产品名称
  product_name: "omnihss"
```

许可证配置

配置项	数据类型	是否必填	默认值
license_server_api_urls	字符串 URL 列表	是	["https://10.0.0.1:8443/api"]
licensee	字符串 许可证持有者名称	是	"ACME Telecom"
product_name	字符串 产品名称	是	"omnihss"

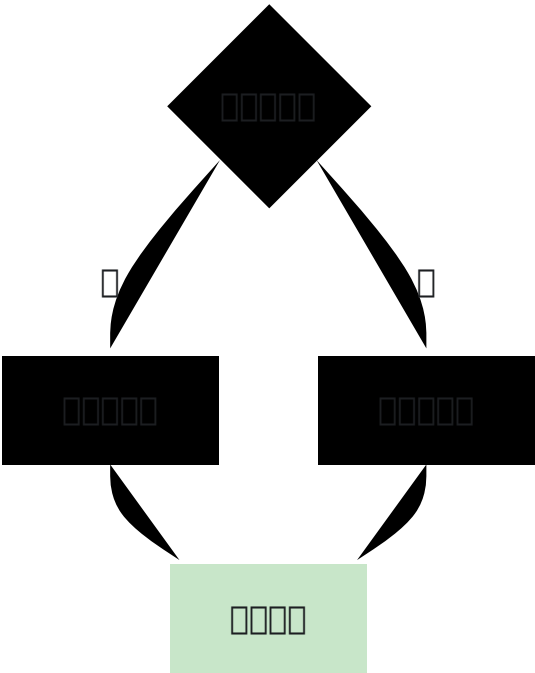
许可证

- 许可证持有者 HSS 名称
- 使用 HTTPS 连接许可证服务器
- 许可证 URL 是否正确配置
- 许可证持有者名称是否正确

□□□□□

□□□□□

□□□□



□□□□□□

OmniHSS □□□□□□□□

- □□□□□□□□□□□□□□
- □□□□ runtime.exs □□□
- □□□□□□□□□□□□□□□□

環境構築

データベース接続

```
# config/runtime.exs

config :hss, Hss.Repo,
  # データベース接続
  username: System.get_env("DATABASE_USERNAME", "root"),
  password: System.get_env("DATABASE_PASSWORD", "password"),
  hostname: System.get_env("DATABASE_HOSTNAME", "localhost"),
  database: System.get_env("DATABASE_NAME", "omnihss"),

  # プールサイズ
  pool_size:
    String.to_integer(System.get_env("DATABASE_POOL_SIZE", "20")),

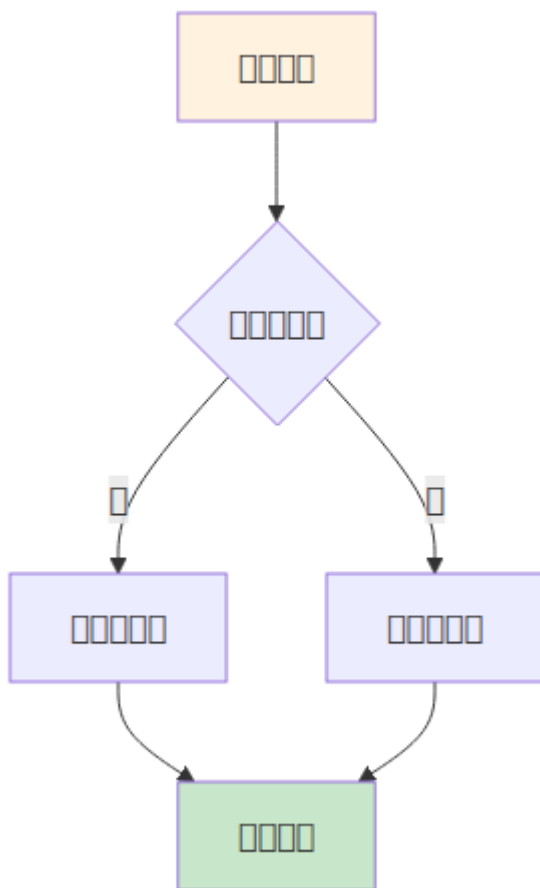
  # タイムアウト
  timeout: 15_000,
  connect_timeout: 15_000,

  # 接続エラー時の挙動
  show_sensitive_data_on_connection_error: false
```

環境変数

変数名	説明	デフォルト値	備考
username	SQL データベースユーザー名	"root"	データベース管理者権限を持つユーザー名
password	SQL データベースパスワード	"password"	データベース管理者パスワード
hostname	SQL データベースホスト名	"localhost"	データベースサーバーの FQDN または IP
database	データベース名	"omnihss"	アプリケーションが使用するデータベース名
pool_size	データベース接続プールサイズ	20	推奨値は10-50程度

□□□□□□□



□□□□□

- □ 20 □□□□□
- □□ "□□□□□" □□
- □□□□□□□□□□□□□□□□ 10
- □□□□□□□ 4MB □□□
- □□□□□□□□□ SQL □□□□□

データベース接続

```
# config/runtime.exs - データベース接続

config :hss, Hss.Repo,
  username: System.fetch_env!("DATABASE_USERNAME"), # データベースユーザー名
  password: System.fetch_env!("DATABASE_PASSWORD"), # データベースパスワード
  hostname: System.get_env("DATABASE_HOSTNAME",
    "db.internal.example.com"),
  database: System.get_env("DATABASE_NAME", "omnihss"),
  port: String.to_integer(System.get_env("DATABASE_PORT",
    "5432")),
  pool_size:
    String.to_integer(System.get_env("DATABASE_POOL_SIZE", "30")),
  ssl: true,
  ssl_opts: [
    cacertfile: "/etc/ssl/certs/postgres-ca.pem",
    verify: :verify_peer
  ]
```

Diameter 配置

Diameter 配置

```
# config/runtime.exs

diameter_config = %{
  service_name: :omnitouch_hss,

  # 配置
  listen_ip: System.get_env("DIAMETER_LISTEN_IP", "10.7.25.186"),
  listen_port:
    String.to_integer(System.get_env("DIAMETER_LISTEN_PORT", "3868")),

  # Diameter 配置
  host: System.get_env("DIAMETER_HOST", "omnihss"),
  realm: System.get_env("DIAMETER_REALM",
    "epc.mnc001.mcc001.3gppnetwork.org"),

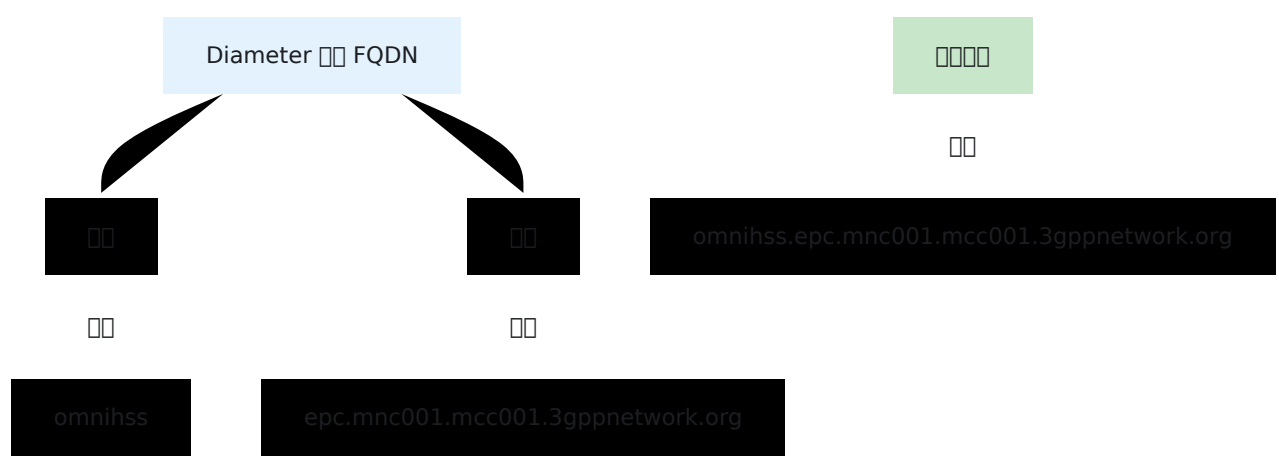
  # 产品信息
  product_name: "OmniHSS",
  vendor_id: 10415, # 3GPP
  supported_vendor_ids: [5535, 10415],

  # 请求超时
  request_timeout: 5000,

  # 对等体
  peers: [
    # 对等体配置
  ]
}

config :hss, :diameter, diameter_config
```

Diameter



Examples

- HSS identifier "omnihss" and "hss01"
- PLMN ID Diameter identifier "epc.mnc001.mcc001.3gppnetwork.org"
- Format {host}.{realm}

Diameter

Examples

```
# config/runtime.exs
```

```
peers: [
```

```
  # MME [] [] [] []
```

```
  %{
```

```
    host: "mme01.epc.mnc001.mcc001.3gppnetwork.org",
```

```
    realm: "epc.mnc001.mcc001.3gppnetwork.org",
```

```
    ip: "10.7.25.100",
```

```
    port: 3868,
```

```
    transport: :sctp, # [] :tcp
```

```
    applications: [:s6a]
```

```
  },
```

```
  # P-GW [] [] [] []
```

```
  %{
```

```
    host: "pgw01.epc.mnc001.mcc001.3gppnetwork.org",
```

```
    realm: "epc.mnc001.mcc001.3gppnetwork.org",
```

```
    ip: "10.7.25.101",
```

```
    port: 3868,
```

```
    transport: :sctp,
```

```
    applications: [:gx]
```

```
  },
```

```
  # I-CSCF [] [] [] []
```

```
  %{
```

```
    host: "icscf01.ims.mnc001.mcc001.3gppnetwork.org",
```

```
    realm: "ims.mnc001.mcc001.3gppnetwork.org",
```

```
    ip: "10.7.25.102",
```

```
    port: 3868,
```

```
    transport: :tcp,
```

```
    applications: [:cx]
```

```
  }
```

```
]
```

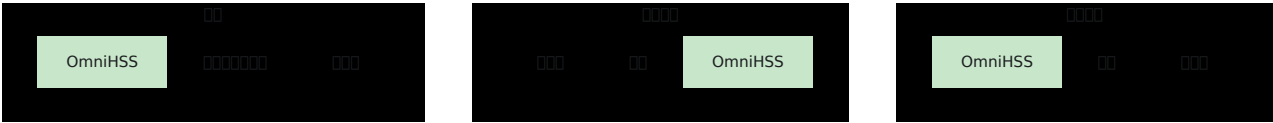
[] [] [] []

[] [] [] [] HSS [] [] [] [] [] []

```
# config/runtime.exs

diameter_config = %{
  # ...
  peers: [] # - 
}
```

Diameter



SCTP			Diameter
TCP			SCTP

PLMN

PLMN

```
# config/runtime.exs

config :hss, :home_plmn, %{
  mcc: System.get_env("HOME_PLMN_MCC", "001"), # 
  mnc: System.get_env("HOME_PLMN_MNC", "001") # 
}
```

HSS 配置

配置 HSS 配置

```
# config/runtime.exs

config :hss,
  # Ecto 配置
  ecto_repos: [Hss.Repo],

  # MME 配置 CLR 配置
  send_clr_on_mme_change: true,

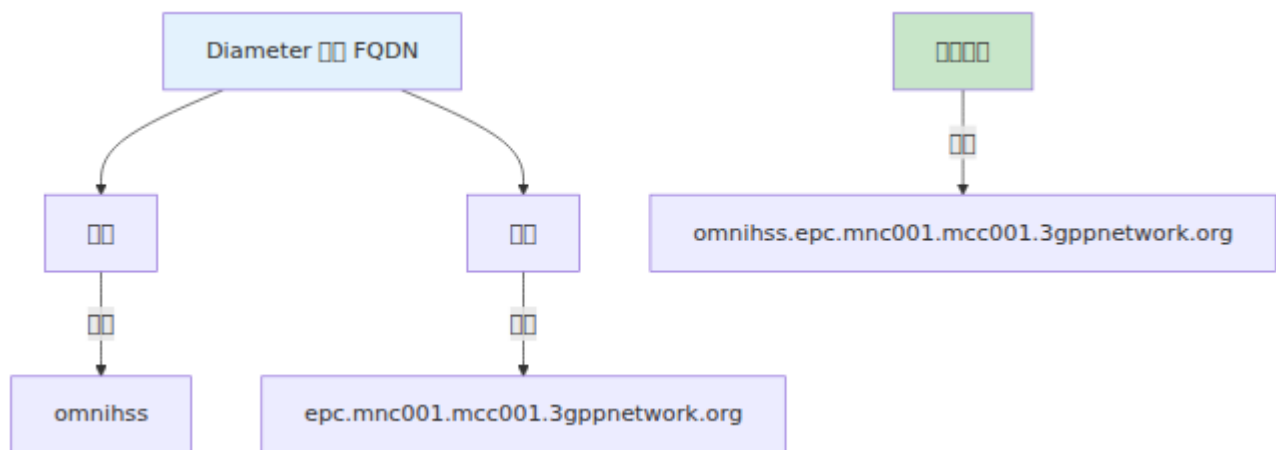
  # 配置 Diameter 配置
  stop_diameter_on_database_failure: true,

  # 配置
  license_enforced: true,
  license_module: LicenseClient
```

HSS 配置

項目	項目	項目	項目
ecto_repos	データベースリポジトリ Ecto リポジトリ	[Hss.Repo]	データベース リポジトリ オブジェクト
send_clr_on_mme_change	MMSEvent MME イベント 送信フラグ	true	MMSEvent リポジトリ オブジェクト 送信フラグ オブジェクト
stop_diameter_on_database_failure	データベースエラー Diameter 停止フラグ	true	データベース リポジトリ オブジェクト 停止フラグ オブジェクト
license_enforced	ライセンス強制	true	ライセンス リポジトリ オブジェクト
license_module	ライセンスモジュール クライアント	LicenseClient	ライセンス リポジトリ

PLMN



PLMN

- AT&T MCC=310 MNC=410
- Verizon MCC=311 MNC=480
- Vodafone MCC=234 MNC=15
- MCC=001 MNC=01

□□□□□□

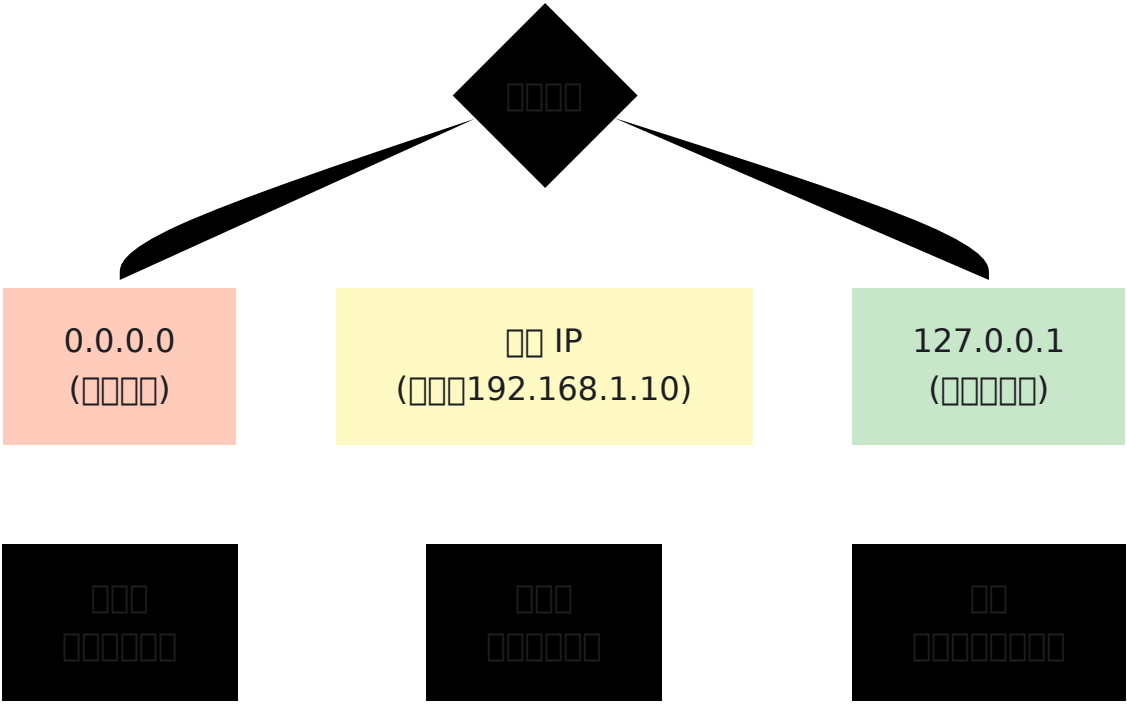
```
# config/runtime.exs

# Diameter □□
listen_ip: System.get_env("DIAMETER_LISTEN_IP", "0.0.0.0"), # □□□
□
# □□□□□□
# listen_ip: "10.7.25.186",

# API □□
config :hss, HssWeb.Api.Endpoint,
  http: [
    ip: {0, 0, 0, 0}, # □□□□
    port: 8443
  ]

# □□□□□□
config :hss, HssWeb.ControlPanel.Endpoint,
  http: [
    ip: {0, 0, 0, 0}, # □□□□
    port: 7443
  ]
```

□□□□□□□□



IMS

S-CSCF

```
# config/runtime.exs

config :hss, :ims, %{
  scscf: %{
    # :random_peer or :round_robin
    selection_method: :random_peer,

    # S-CSCF peers
    peers: [
      %{
        host:
        "sip:scscf01.ims.mnc001.mcc001.3gppnetwork.org:5060",
        capabilities: [] # capabilities
      },
      %{
        host:
        "sip:scscf02.ims.mnc001.mcc001.3gppnetwork.org:5060",
        capabilities: []
      }
    ]
  }
}
```

S-CSCF

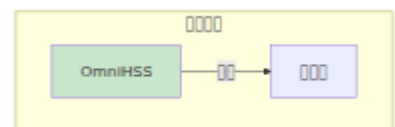
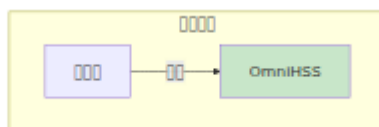
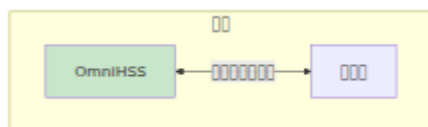


Diagram 4: S-CSCF (green box) connected to HSS (blue box) via a bidirectional arrow. The connection is labeled with the text "HSS".

Peer	Service	Mode
:random_peer	S-CSCF	Random
:round_robin	S-CSCF	Round Robin

IMS

IMS EPC

```
# EPC
"epc.mnc001.mcc001.3gppnetwork.org"

# IMS
"ims.mnc001.mcc001.3gppnetwork.org"
```

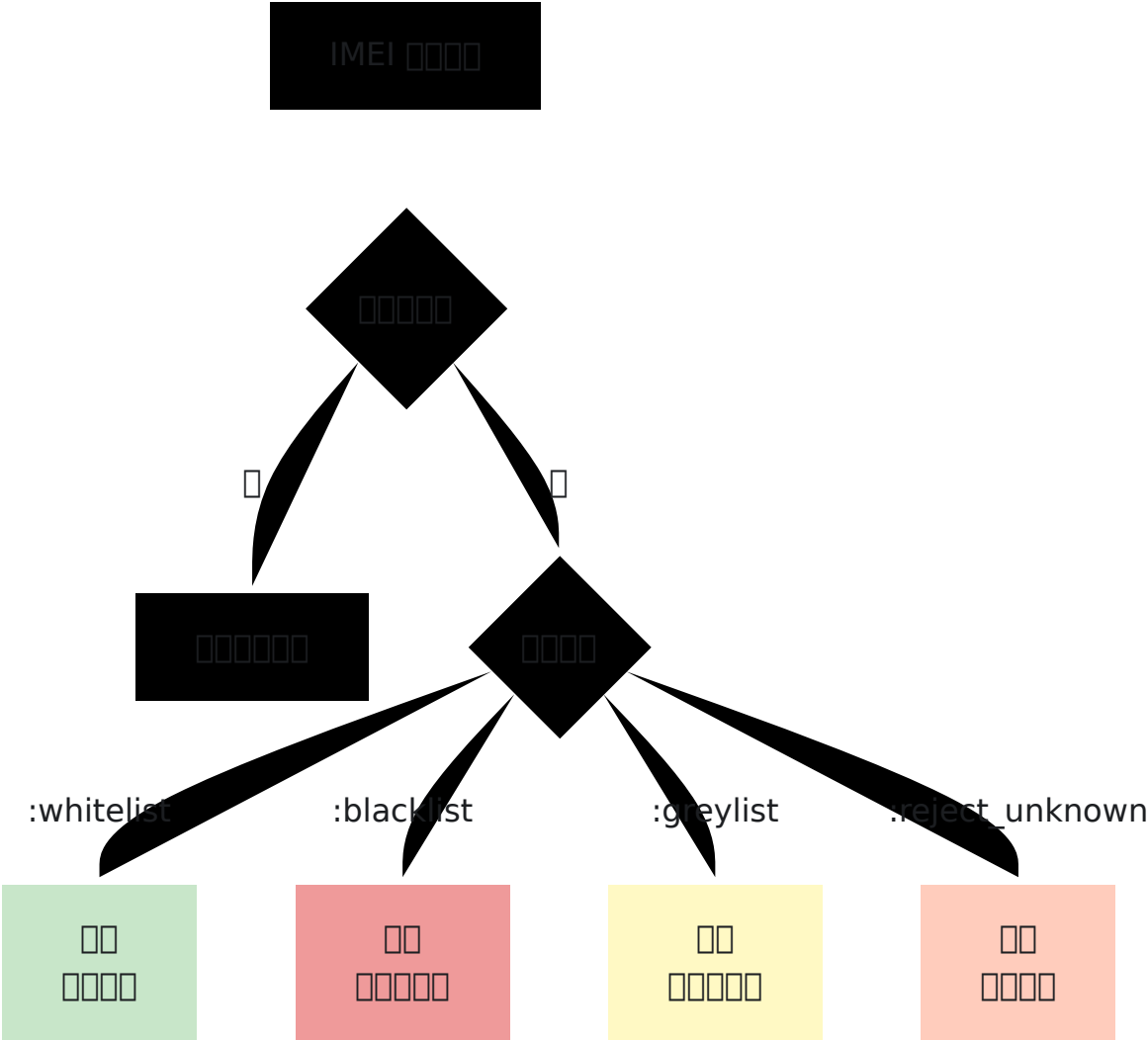
EIR

EIR

```
# config/runtime.exs

config :hss, :eir, %{
  #
  unknown_equipment_behaviour: :whitelist
  #
  # :whitelist -
  # :blacklist -
  # :greylist -
  # :reject_unknown_equipment -
}
```

IMEI 黑名单



黑名单

黑名单	黑名单	黑名单
:whitelist	黑名单 IMEI	黑名单
:blacklist	黑名单 IMEI	黑名单
:greylist	黑名单 IMEI	黑名单
:reject_unknown_equipment	黑名单	黑名单

黑名单 :whitelist 黑名单 :greylist 黑名单 :blacklist

API 環境構築

API 環境

```
# config/config.exs

config :hss, HssWeb.Api.Endpoint,
  url: [host: "localhost"],
  render_errors: [view: HssWeb.ErrorView, accepts: ~w(json)],
  pubsub_server: Hss.PubSub,

# HTTPS 環境
https: [
  port: 8443,
  cipher_suite: :strong,
  certfile: "priv/cert/omnitouch.crt",
  keyfile: "priv/cert/omnitouch.pem"
]
```

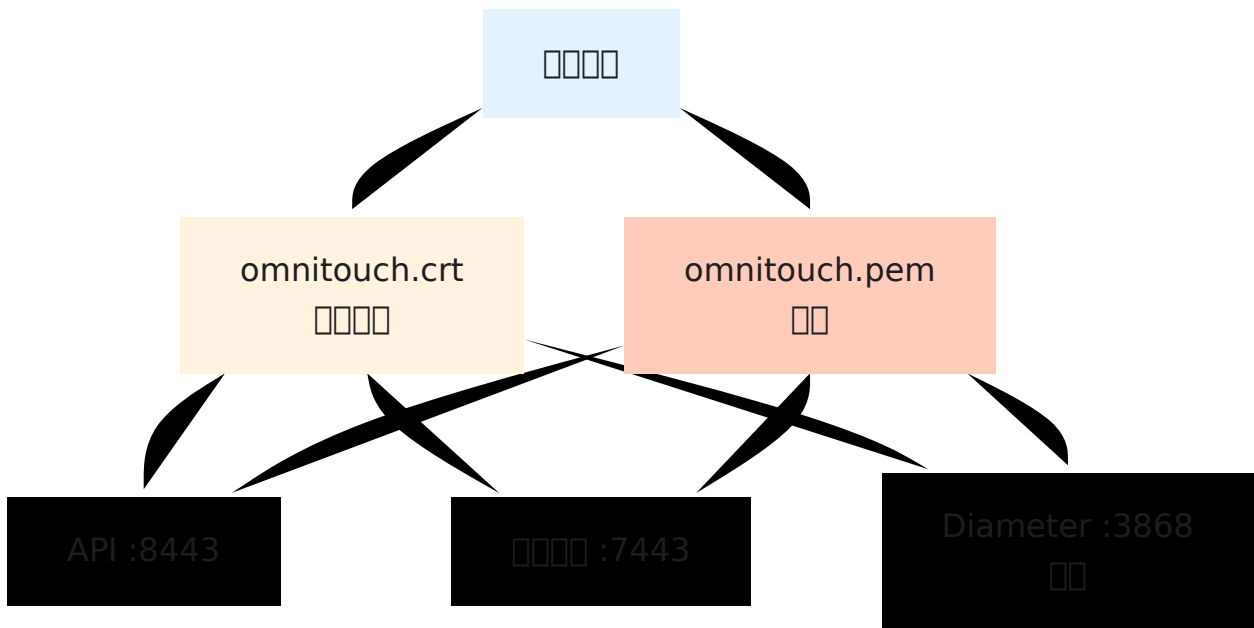
環境構築

```
# config/config.exs

config :hss, HssWeb.ControlPanel.Endpoint,
  url: [host: "localhost"],
  render_errors: [view: HssWeb.ErrorView, accepts: ~w(html json)],
  pubsub_server: Hss.PubSub,
  live_view: [signing_salt: "some-secret"],

# HTTPS 環境
https: [
  port: 7443,
  cipher_suite: :strong,
  certfile: "priv/cert/omnitouch.crt",
  keyfile: "priv/cert/omnitouch.pem"
]
```


TLS 証明書



証明書

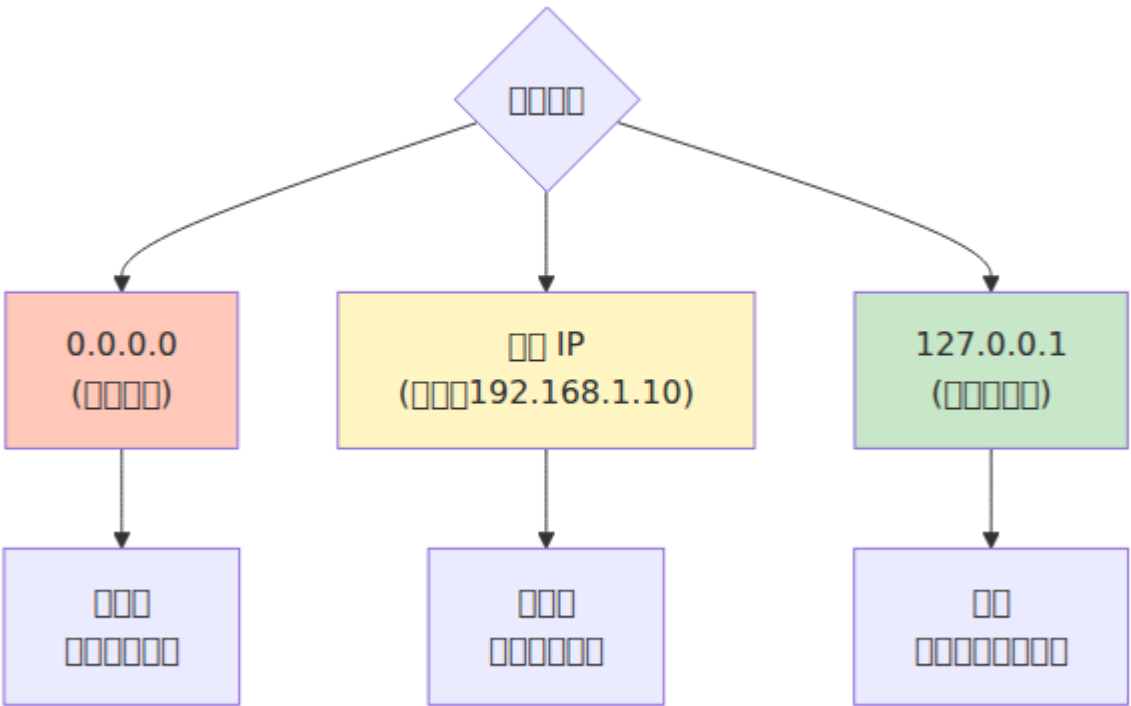
- 証明書 X.509 形式
- 証明書
- 証明書
- CN と SAN 証明書

証明書

```
https: [  
  port: 8443,  
  cipher_suite: :strong,  
  certfile: System.get_env("TLS_CERT_FILE",  
    "/etc/ssl/certs/omnihss.crt"),  
  keyfile: System.get_env("TLS_KEY_FILE",  
    "/etc/ssl/private/omnihss.key"),  
  cacertfile: System.get_env("TLS_CA_FILE", "/etc/ssl/certs/ca-  
    bundle.crt")  
]
```

□□□□□□

□□□□□□



□□□□□□

□□□□

- ☐ □□□□□□□□□□□□□□
- ☐ □□ PLMN□MCC□MNC□
- ☐ Diameter □□□□□
- ☐ Diameter □□ IP □□□
- ☐ API □□□□□□ TLS □□
- ☐ □□□□□□□□□□□□ URL□□□□□□□□□□□□□□
- ☐ HSS □□□□□□send_clr_on_mme_change□stop_diameter_on_database_failure□

□□□□□□

- ☐ □□ Diameter □□□□□□□□□□□□□□
- ☐ □□□□□□□□ Diameter □□□□□□ 3868□
- ☐ □□□□□□□□ HTTPS □□□□□□ 7443□8443□

☐ Diameter 設定 DNS 設定

IMS 設定 IMS 設定

☐ S-CSCF 設定

☐ S-CSCF 設定

☐ IMS 設定

設定

☐ EIR 設定

☐ 設定

☐ 設定

設定

設定

1. 設定

```
設定
```

2. 設定

```
https://[hostname]:7443
設定
```

3. API 設定

```
curl -k https://[hostname]:8443/api/status
```

4. Diameter 設定

```
Diameter 設定
設定
```

5. 資料庫

資料庫系統
資料庫 SQL 查詢

配置文件

```
# config/runtime.exs - 数据库配置
```

```
import Config
```

```
#
```

```
=====
```

```
# 数据库配置
```

```
#
```

```
=====
```

```
config :hss, Hss.Repo,
```

```
  username: System.fetch_env!("DATABASE_USERNAME"),
```

```
  password: System.fetch_env!("DATABASE_PASSWORD"),
```

```
  hostname: System.get_env("DATABASE_HOSTNAME", "db.omnihss.internal"),
```

```
  database: System.get_env("DATABASE_NAME", "omnihss"),
```

```
  port: String.to_integer(System.get_env("DATABASE_PORT", "5432")),
```

```
  pool_size: String.to_integer(System.get_env("DATABASE_POOL_SIZE", '10')),
```

```
  timeout: 15_000,
```

```
  connect_timeout: 15_000,
```

```
  ssl: true,
```

```
  ssl_opts: [
```

```
    cacertfile: "/etc/ssl/certs/postgres-ca.pem",
```

```
    verify: :verify_peer
```

```
  ]
```

```
#
```

```
=====
```

```
# 许可证配置
```

```
#
```

```
=====
```

```
config :license_client,
```

```
  license_server_api_urls: [System.get_env("LICENSE_SERVER_URL",  
"https://license.example.com:8443/api")],
```

```
  licensee: System.get_env("LICENSE_ORGANIZATION", "HSS"),
```

```
  product_name: "omnihss"
```

```
#
```

```
=====
```

```
# 配置 PLMN 和 HSS 连接
```

```
#
```

```
=====
```

```

config :hss,
  ecto_repos: [Hss.Repo],
  home_plmn: %{
    mcc: System.get_env("HOME_PLMN_MCC", "001"),
    mnc: System.get_env("HOME_PLMN_MNC", "001")
  },
  send_clr_on_mme_change: true,
  stop_diameter_on_database_failure: true,
  license_enforced: true,
  license_module: LicenseClient

#
=====
# Diameter
#
=====
diameter_config = %{
  service_name: :omnitouch_hss,
  listen_ip: System.get_env("DIAMETER_LISTEN_IP", "10.7.25.186"),
  listen_port: String.to_integer(System.get_env("DIAMETER_LISTEN_PORT", "3868")),
  host: System.get_env("DIAMETER_HOST", "omnihss01"),
  realm: System.get_env("DIAMETER_REALM", "epc.mnc001.mcc001.3gppnetwork.org"),
  product_name: "OmniHSS",
  vendor_id: 10415,
  supported_vendor_ids: [5535, 10415],
  request_timeout: 5000,
  peers: [
    %{
      host: "mme01.epc.mnc001.mcc001.3gppnetwork.org",
      realm: "epc.mnc001.mcc001.3gppnetwork.org",
      ip: "10.7.25.100",
      port: 3868,
      transport: :sctp,
      applications: [:s6a]
    }
  ]
}

config :hss, :diameter, diameter_config

#
=====

```

```

# IMS []
#
=====
config :hss, :ims, %{
  scscf: %{
    selection_method: :random_peer,
    peers: [
      %{host: "sip:scscf01.ims.mnc001.mcc001.3gppnetwork.org:5060"},
      %{host: "sip:scscf02.ims.mnc001.mcc001.3gppnetwork.org:5060"}
    ]
  }
}

#
=====
# EIR []
#
=====
config :hss, :eir, %{
  unknown_equipment_behaviour: :whitelist
}

#
=====
# API []
#
=====
config :hss, HssWeb.Api.Endpoint,
  http: [ip: {0, 0, 0, 0}, port: 8443],
  https: [
    port: 8443,
    cipher_suite: :strong,
    certfile: System.get_env("TLS_CERT_FILE", "/etc/ssl/certs/omnihss"),
    keyfile: System.get_env("TLS_KEY_FILE", "/etc/ssl/private/omnihss"),
  ],
  url: [host: System.get_env("API_HOST", "api.omnihss.internal"), port: 8443]

#
=====
# []
#
=====
config :hss, HssWeb.ControlPanel.Endpoint,

```

```
http: [ip: {0, 0, 0, 0}, port: 7443],
https: [
  port: 7443,
  cipher_suite: :strong,
  certfile: System.get_env("TLS_CERT_FILE", "/etc/ssl/certs/omnihss"),
  keyfile: System.get_env("TLS_KEY_FILE", "/etc/ssl/private/omnihss"),
],
url: [host: System.get_env("CP_HOST", "hss.omnihss.internal"), port
```

← 000000 | 00000000 →

OmniHSS 開箱開箱開箱

← 開箱開箱

開箱

- 開箱開箱
- 開箱開箱
- 開箱
- 開箱
- 開箱
- 開箱
- 開箱開箱

開箱開箱開箱

OmniHSS 開箱開箱開箱開箱開箱開箱開箱開箱開箱開箱 Phoenix LiveView
開箱開箱開箱開箱開箱開箱

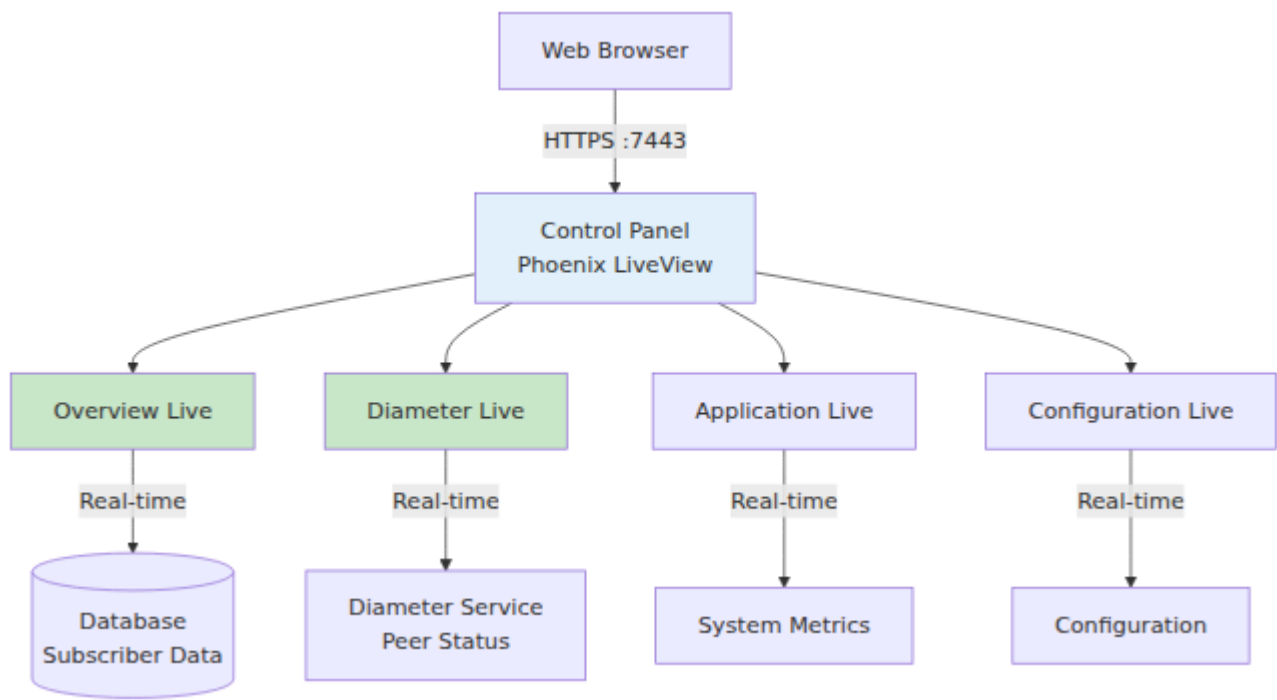
開箱開箱

- 開箱 - 開箱開箱
- 開箱 - 開箱開箱開箱開箱
- 開箱 - 開箱開箱開箱
- 開箱 - 開箱開箱
- 開箱開箱 - 開箱開箱

□□□□

URL: https://[hostname]:7443
Protocol: HTTPS Only
Port: 7443 (configurable)
Certificate: Configured in config/config.exs

□□□□□□



□□□□□□□

□□□□

1. □□□□□□□□□□
2. □□□ `https://[hostname]:7443`
3. □□ TLS □□□□□□□□□□□□
4. □□□□□□□□□□□□□□□□

TLS 简介

安全传输层协议 (TLS) 是用于在两个通信应用程序之间提供私密性和数据完整性的一种安全协议。

它通过加密数据来防止窃听和攻击者篡改数据流。

简介

- 默认 **7443** 端口进行通信
- **HTTPS** 协议 - 基于 HTTP
- 默认端口 7443

支持

支持 LiveView, WebSockets

- Chrome/Chromium
- Firefox
- Safari
- Edge

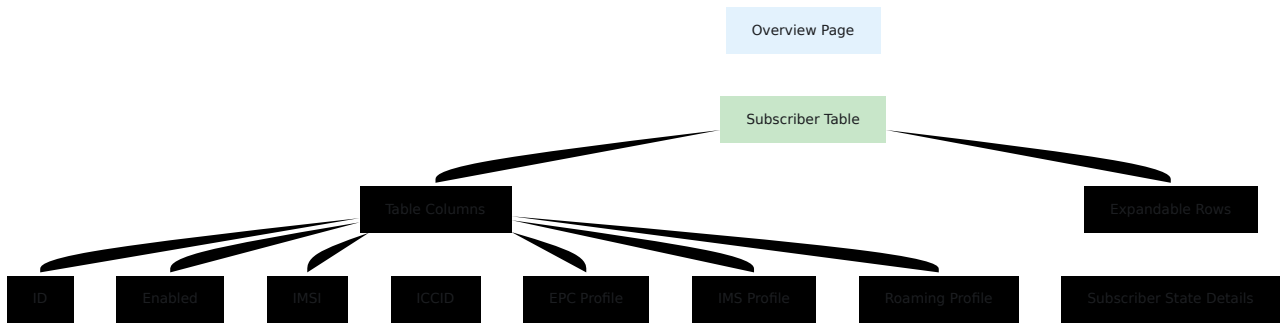
不支持 Internet Explorer

访问

URL: `https://[hostname]:7443/overview`

默认端口 7443

Table



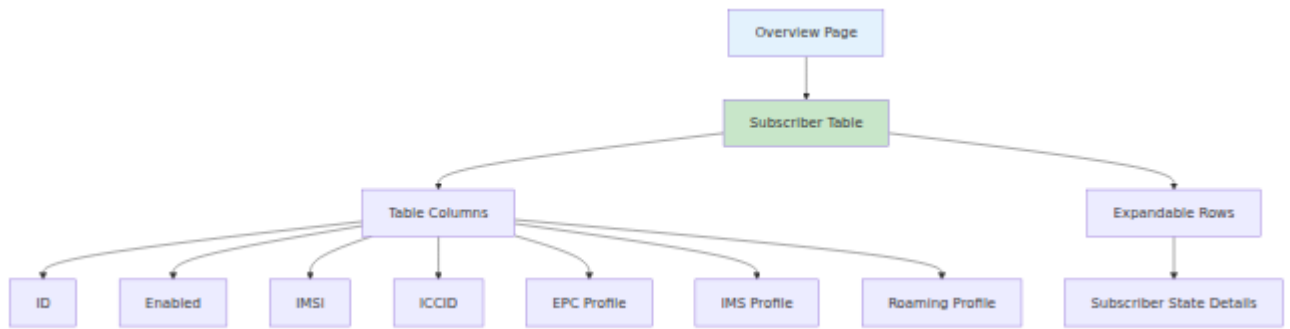
Table

Field	Description	Notes
ID	Subscriber ID	UUID
Enabled	Subscription Status	✓ (Active) / ✗ (Inactive)
IMSI	International Mobile Subscriber Identity	14-15 Digits
ICCID	Integrated Circuit Card ID	19-20 Digits "N/A"
EPC Profile	Evolved Packet Core Profile	Profile ID
IMS Profile	IP Multimedia Subsystem Profile	Profile ID or "N/A"
Roaming Profile	Roaming Profile	Profile ID or "N/A"

Table

Subscriber State Details

Table



□□□

- **MCC** - □□□□□□□3 □□□□
- **MNC** - □□□□□□□2-3 □□□□
- **TAC** - □□□□□□
- **Cell ID** - □□□□□□□□
- **eNodeB ID** - □□□□□□
- **ECI** - E-UTRAN □□□□□□

□□□□□

□□□□

- □□□□□□ **MME** - □□□□□□ MME □□□□
- □□□□□□□□ - MME □□□□□□
- **RAT** □□ - □□□□□□□□□□“E-UTRAN”□□ LTE□
- □□□□□□□□ - □□□□□□□□□□□□

IMS □□

□□□□

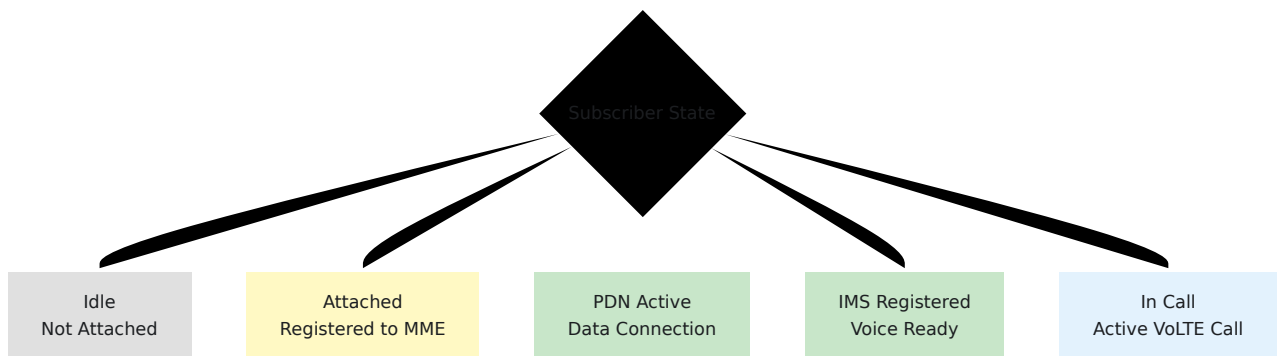
- □□□□ **S-CSCF** - □□□□□□ S-CSCF SIP URI
- **IMS** □□□□ - SIP URI□□□□sip:[+14155551234@ims.example.com](mailto:sip:+14155551234@ims.example.com)□
- □□□□□□ **P-CSCF** - □□□□□□□□ HSS □ P-CSCF
- □□□□□□ **I-CSCF** - □□□□□□□□ HSS □ I-CSCF

□□□□□

□□□□

- **PDN** 設定 - 設定可能なPDN
- 設定 - 設定 VoLTE 設定

設定



設定

- **Idle:** 設定 MME
- **Attached:** 設定 MME 設定
- **PDN Active:** PDN 設定 > 0
- **IMS Registered:** 設定 S-CSCF 設定
- **In Call:** 設定 > 0

設定

設定 **1** 設定

設定

- 設定
- 設定
- 設定

設定

1. 設定

- 設定
- 設定

- IMS 認證

2. 認證

- 認證流程
- 認證消息交互
- 認證結果
- 認證失敗

3. 認證

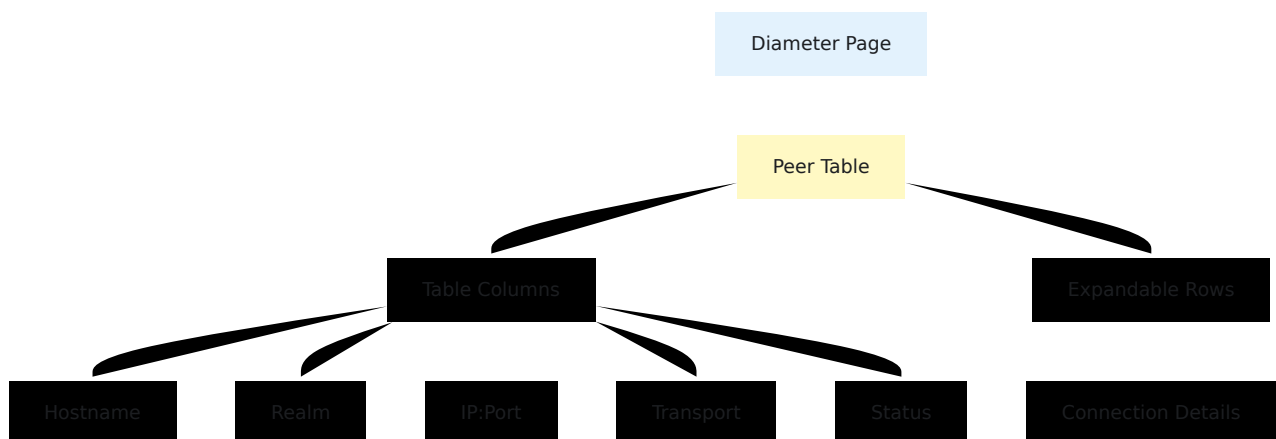
- 認證流程
- PDN 認證
- VoLTE 認證

認證

URL: `https://[hostname]:7443/diameter`

認證消息交互

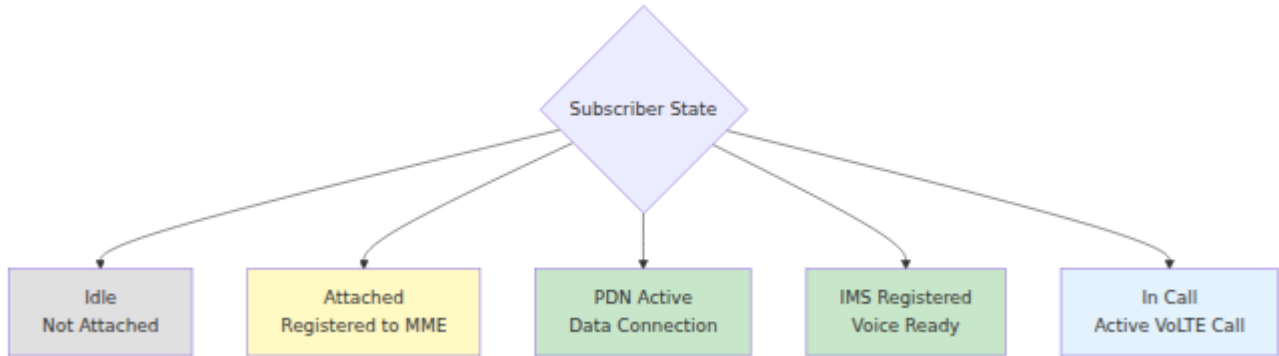
認證



Table

Field	Value	Description
Hostname	String	FQDN
Realm	String	IP
IP:Port	String	IP Address
Transport	String	TCP or SCTP
Status	String	Connected / Disconnected

Subscriber State



Subscriber State

Subscriber State

Subscriber State

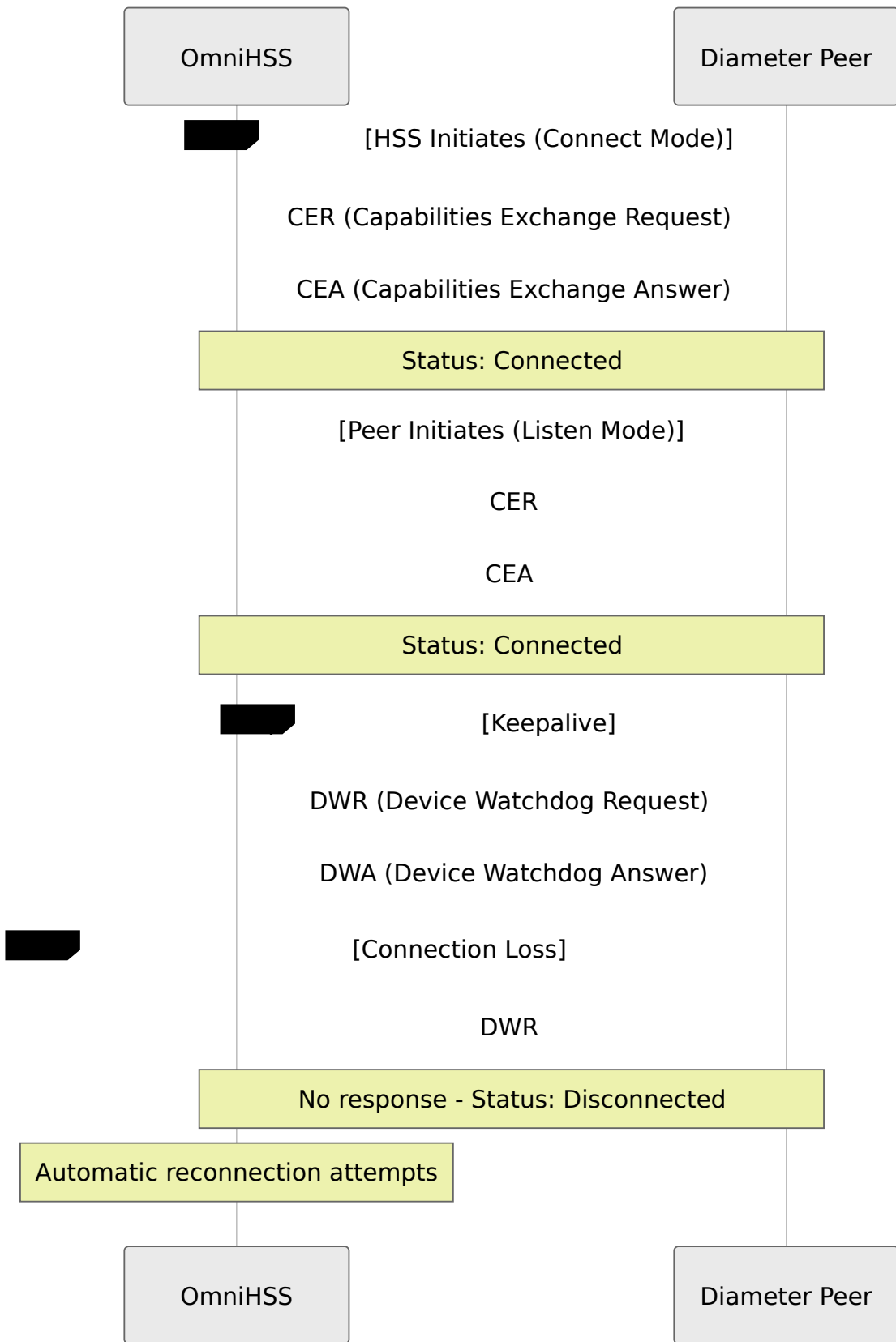
- Subscriber - HSS Subscriber
- Subscriber - Subscriber
- Subscriber ID - Subscriber

Subscriber ID

- 16777251 - S6a (MME)

- 16777238 - Gx (P-GW)
- 16777216 - Cx (I-CSCF, S-CSCF)
- 16777217 - Sh (□□□□□)
- 16777236 - Rx (P-CSCF)
- 16777252 - S13 (EIR □□□□□□□□)

□□□□□□



目標

目標 1 は 目標 2

目標

1. 目標

- 目標 1.1
- 目標 1.2
- 目標 1.3

2. 目標

- 目標 2.1
- 目標 2.2 TCP と SCTP
- 目標 2.3 ID 番号
- 目標 2.4

3. 目標

- 目標 3.1
- 目標 3.2
- 目標 3.3

目標

目標 1.1

目標 1.2

1. 目標

2. 目標

3. 目標

4. 目標

5. 目標 TLS

目標 1.1

1. ping [peer-ip]
2. telnet [peer-ip] 3868
- 3.
4. HSS
5. HSS

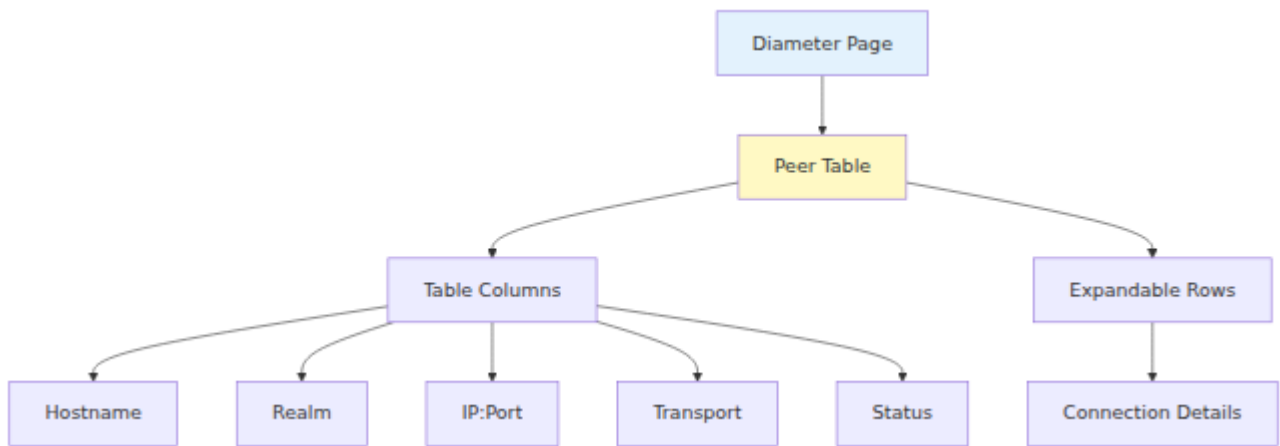
- 1.
- 2.
- 3.
- 4.

- 1.
- 2.
- 3.
4. ID

URL: https://[hostname]:7443/application

- Erlang VM
-
- OmniHSS
- **Erlang VM**

□□□□



□□

1. □□□□

- □□□□□□□□
- □□□□□□□□□□□□□□
- □□□□□□□□

2. □□□□

- □□□□□□□□
- □□□□□□□□□□
- □□□□□□□□□□

3. □□□□

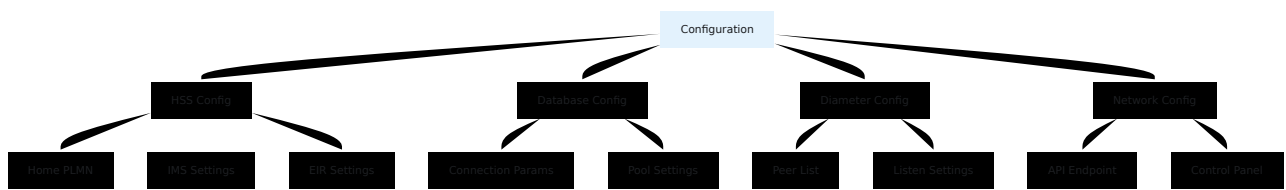
- □□□□□□
- □□□□□□□□
- □□ Erlang VM □□

□□□□

URL: `https://[hostname]:7443/configuration`

OmniHSS

-
-
-



1.

-
-
-

2.

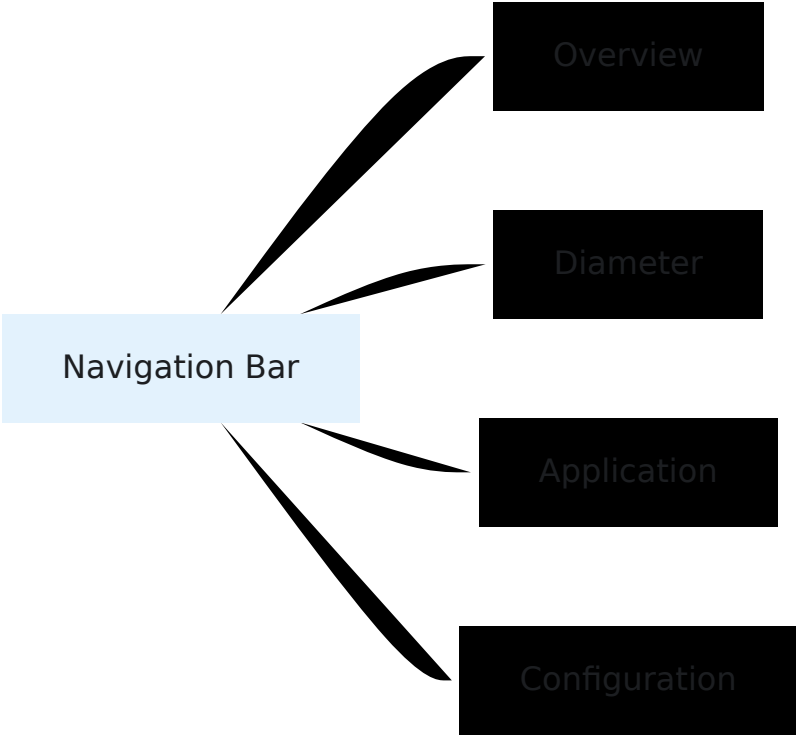
-
-
-

3.

-
-

□□□□□

□□□□□



□□□□□□□□□□□□□□□□□□

□□□□□

□□□□□□□□□□□□□□□□□□□□□□

- **Ctrl+R / F5** - □□□□□□□□□□□□□□□□
- **Ctrl+F** - □□□□□□
- **Ctrl+T** - □□□□□□□□□□□□□□

□□□□□

□□□□□□□□□□□□□□□□□□□□□□□□

□□□□

- □□ 1□□□□□□□□□□□□
- □□ 2□□□□□□□□□□□□

- 3D

1920x1080

1.

-
-
-

2.

-
-
-

3.

-
-
-

□□□□□□□□

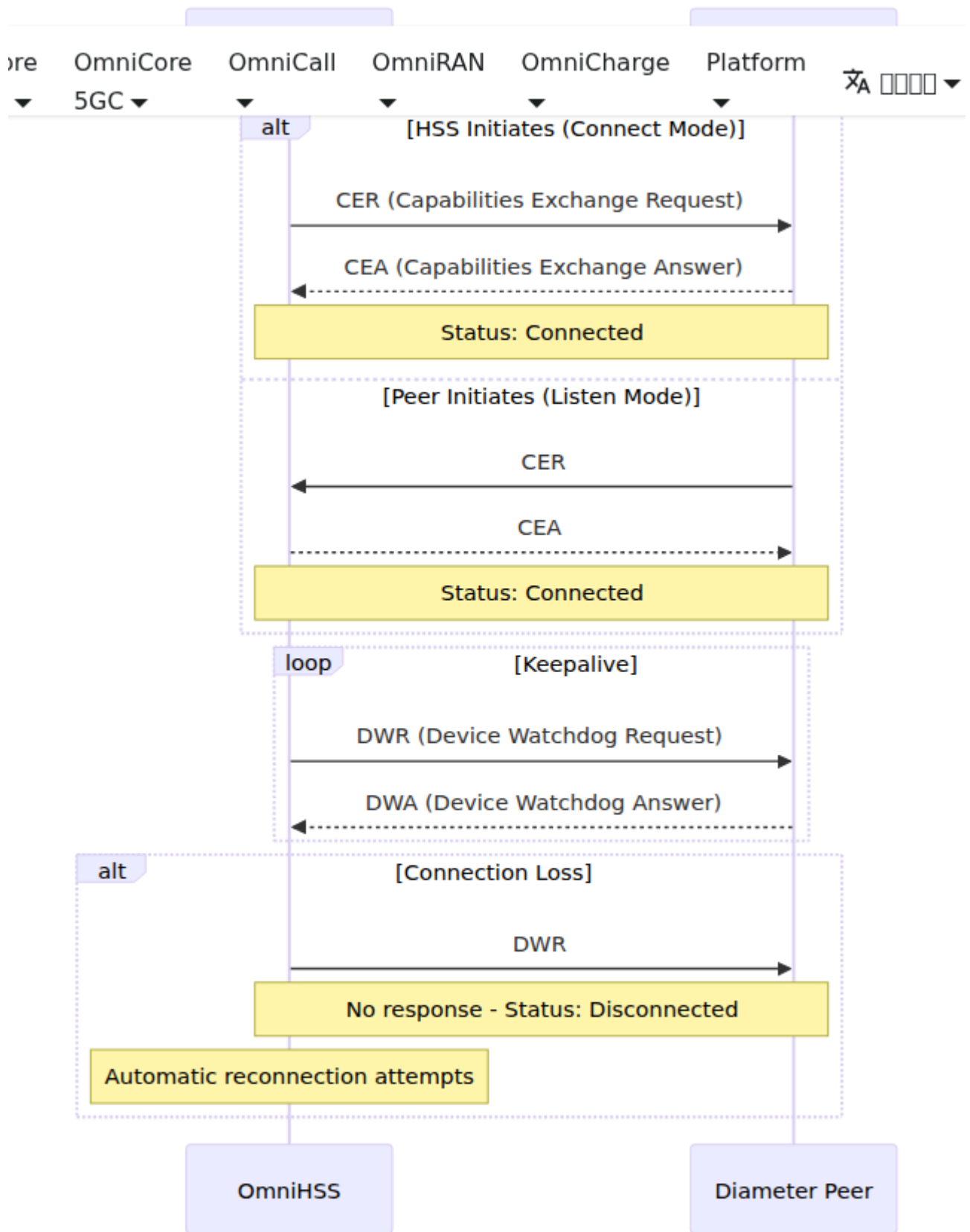


Table 1

Table 1: Performance Metrics

Category	Metric A	Metric B
Overall Score	1.5	2+ (Excellent)
Efficiency	> 80%	> 90%
Reliability	> 5%	> 10%
Accuracy	> 80% (High)	> 95% (Very High)

--	--	--	--	--

← ☐ ☐ ☐ ☐ ☐ ☐

11

- ☐ ☐
- ☐ ☐ ☐ ☐
- ☐ ☐ ☐ ☐
- ☐ ☐ ☐ ☐
- ☐ ☐ ☐ ☐
- ☐ ☐
- ☐ ☐ ☐ ☐

11

OmniHSS 与 PostgreSQL 的兼容性测试/验证报告

□ □ □ □

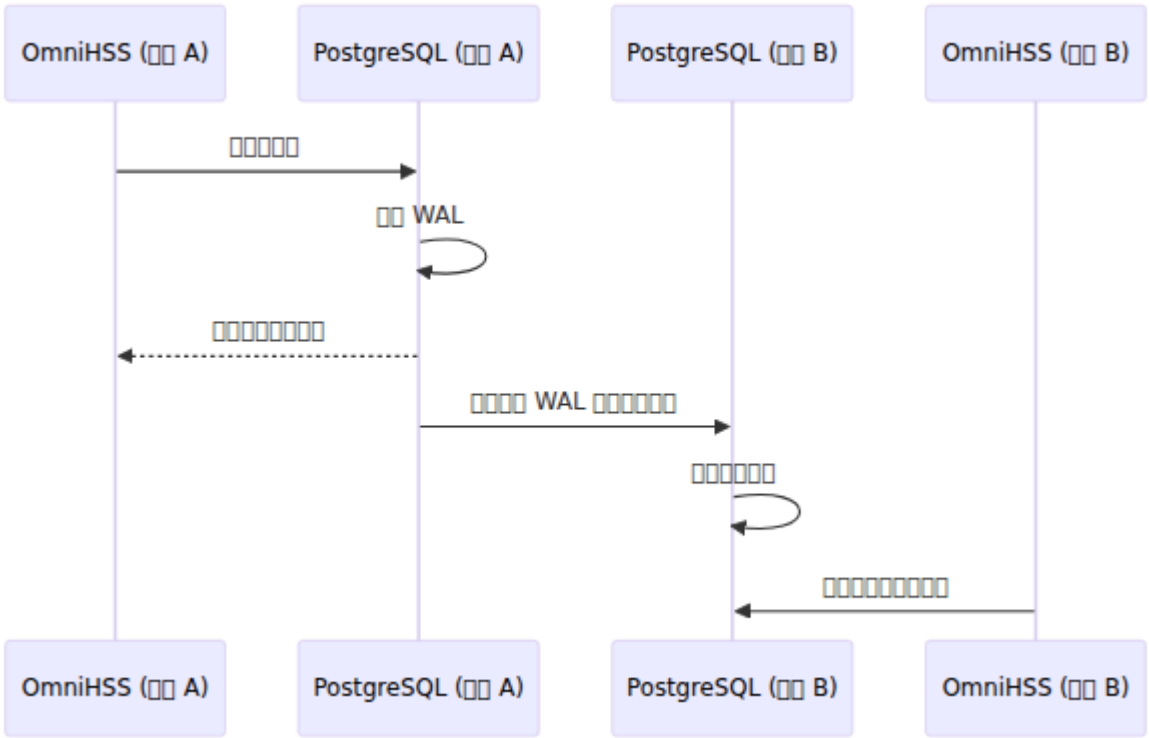
- 00000000/00000000000000000000000000000000
- 00000000 PostgreSQL WAL000000000000000000000000
- 00000000000000000000000000000000 WAL 000000000000000000
- 00000000000000000000000000000000   000000000000000000
- **UUID** 00000000 UUIDv4 0000000000 ID 00000000

データベース

	DB	データベース
データベース	データベース	データベース
データベース	データベース	データベース/DB
データベース	データベース	DB
データベース	データベース	データベースデータベース
データベース	データベースデータベース	DB WAL データベース

データベース

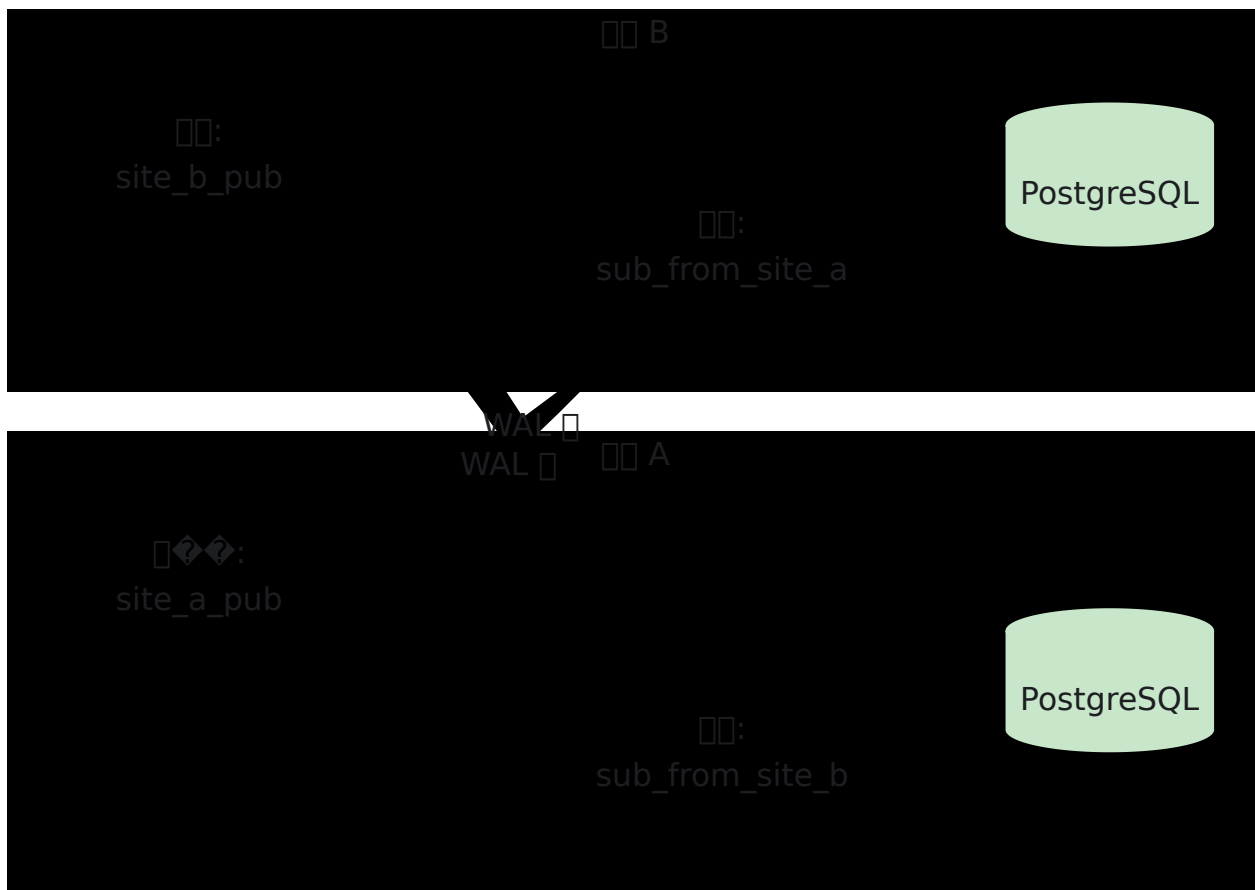
データベース



主/从复制

主从复制

- 主 复制槽名称为 `schema_migrations`
- 主 复制槽名称为



主从复制 WAL 槽

主从复制槽

1. 复制槽名称
2. WAL 槽名称
3. 复制槽名称
4. 复制槽名称 WAL 槽名称
5. 复制槽名称

WAL 槽 `max_slot_wal_keep_size` 复制槽名称

CREATE TABLE

UUID PRIMARY KEY NOT NULL)INSERT INTO

origin = none

schema_migrations

CREATE INDEX

subscriber_state pdn_session lte_call Diameter

ULR MME CCR PGW AAR

subscriber_state MME

OmniHSS LWW

```
CREATE TRIGGER lww_subscriber_state_trigger
BEFORE UPDATE ON subscriber_state
FOR EACH ROW
EXECUTE FUNCTION lww_subscriber_state();
```

updated_at

- A B B
- A
-

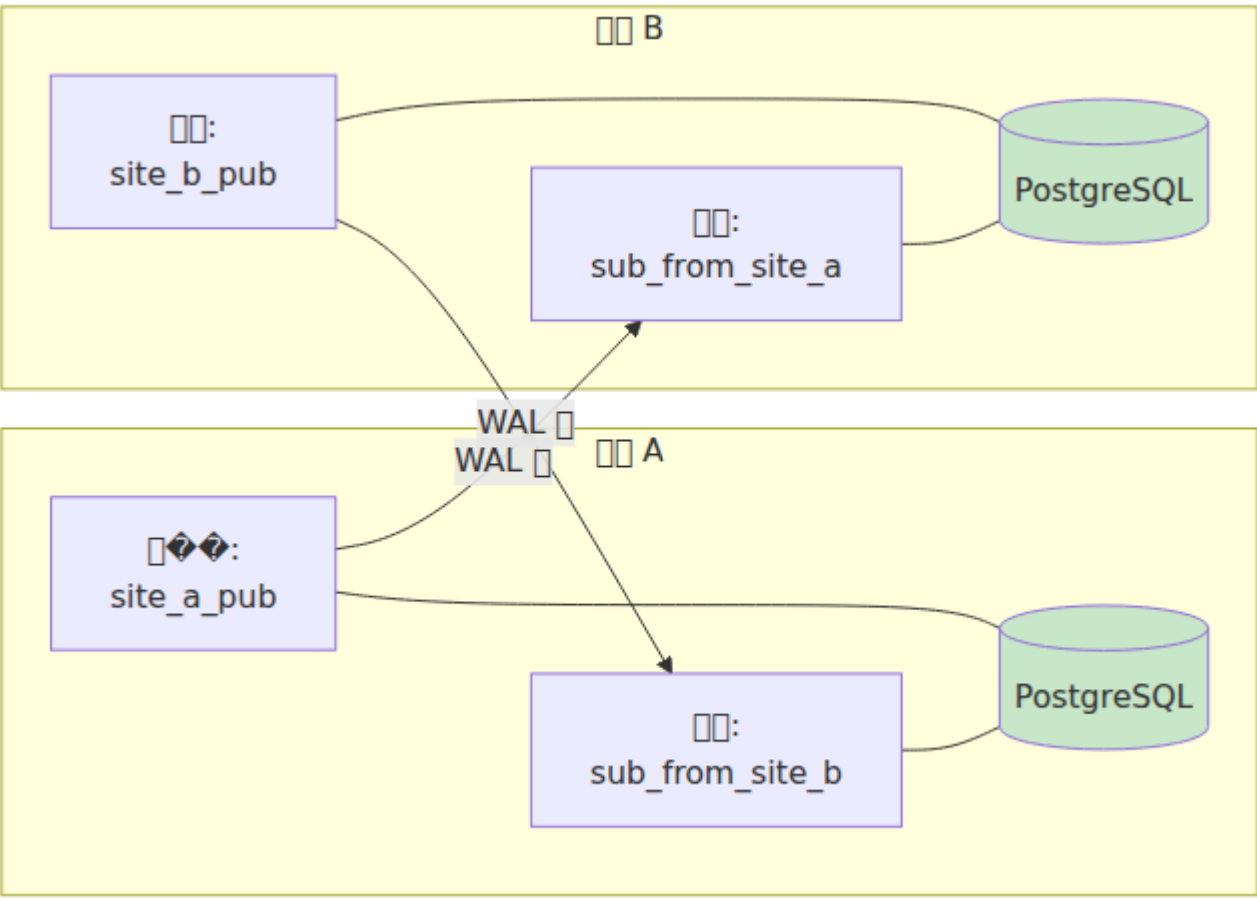
LWW

項目	項目名	項目説明
subscriber_state	ULR, SAR, AIR, PUR	状態遷移/状態
pdn_session	CCR-I, CCR-T	PDN セッション開始/終了
lte_call	AAR, STR	VoLTE 状態遷移/状態

データベーススキーマはAPNごとにLWWAPIで管理されています

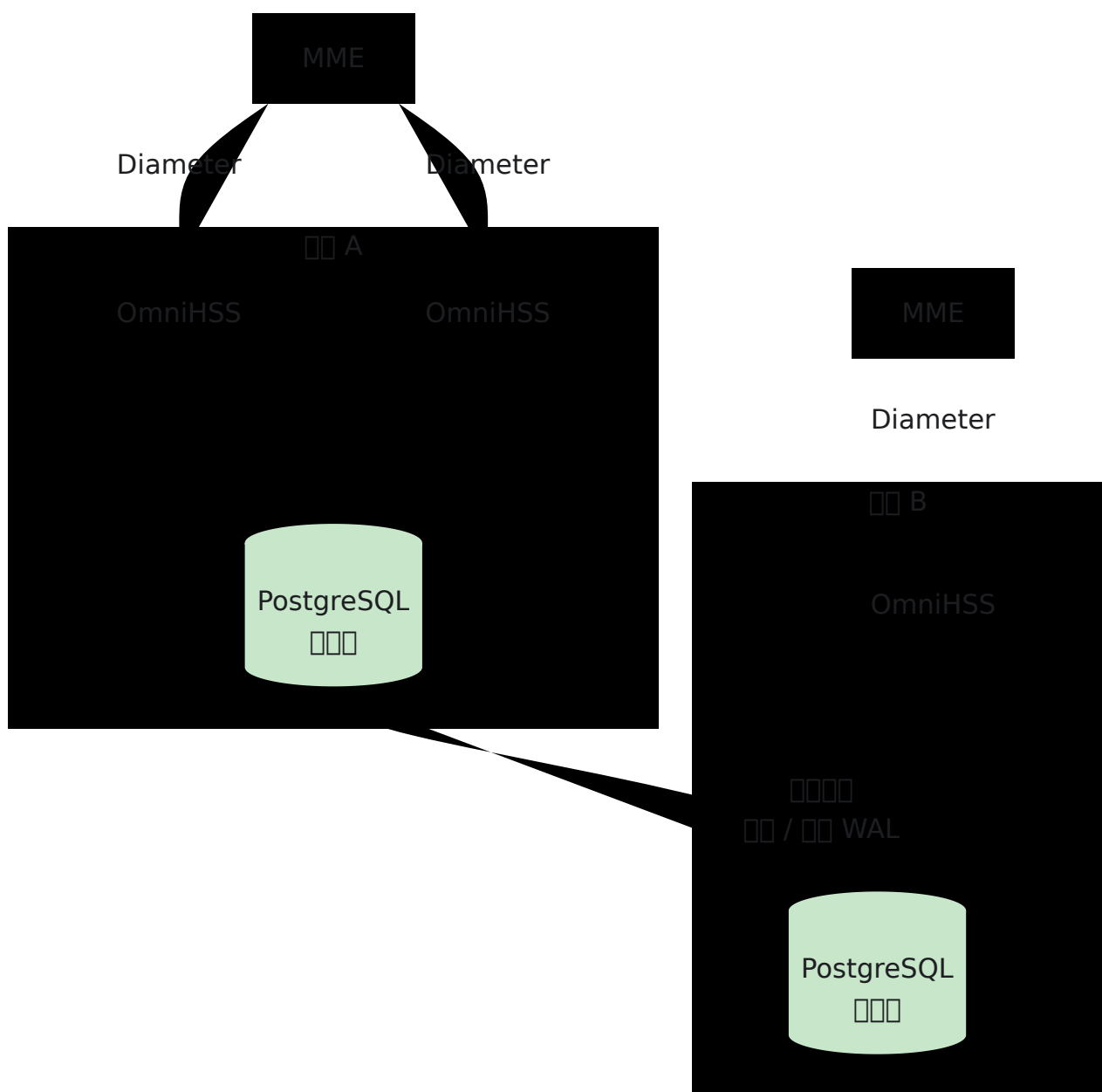
データベース

データベーススキーマ



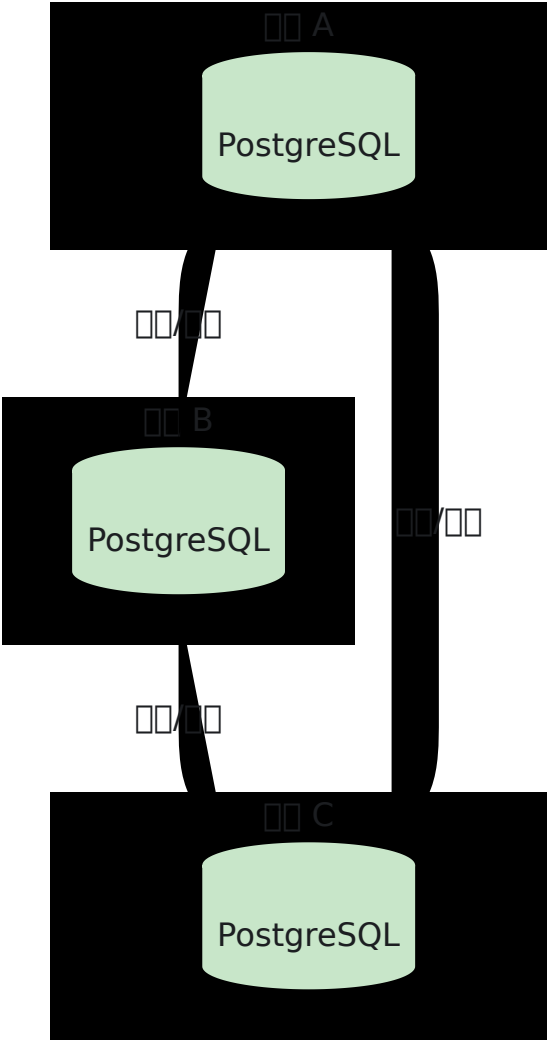
Site HSS データベース PostgreSQL データベース PostgreSQL データベース

--	--	--	--	--



MME HSS
 Diameter

□□□□□



□□□□□□□□□□□□□□□□□□□□ Ansible □□□□□□□□□□□□□□□□□□□□ hss □□□□□□□□□□

□□□□□

□□	□□	□□
5432	TCP	PostgreSQL □□□□□□□□□□

□□□□

Ansible □□

□□□□□□□□□□ HSS □□□□□

```
hss:
  hosts:
    site-a-hss01:
      ansible_host: 10.80.12.140
    site-a-hss02:
      ansible_host: 10.80.12.141
  vars:
    omnihss:
      database_type: postgres
      database_host: localhost
      database_port: 5432
      database_name: omnihss
      database_username: omnihss
      database_password: "secure_password"
      replication:
        enabled: true
```

Ansible 配置

配置项	数据类型	默认值	配置项	配置项
database_type	字符串	postgres	数据库类型	postgres
database_host	字符串	localhost	PostgreSQL 数据库 Postgres 数据库 OmniHSS 数据库	localhost
database_port	字符串	5432	PostgreSQL 端口	
database_name	字符串	omnihss	数据库名称	
database_username	字符串	-	PostgreSQL 用户名	CREATEDB REPLICATION
database_password	字符串	-	PostgreSQL 密码	
replication.enabled	布尔值	false	是否启用复制	true

配置项

配置项说明

- 配置项 hss Ansible 配置项

2. `prod_master_peers.yml` `connected_sites.hss`

`hss` `connected_sites.hss`

(prod_master_peers.yml)

```
# group_vars/prod_master_peers.yml
connected_sites:
  hss:
    - remote-site-hss01: 10.80.20.140
    - remote-site-hss02: 10.80.20.141
```

PostgreSQL

Ansible PostgreSQL

<code>wal_level</code>	<code>logical</code>	WAL
<code>max_wal_senders</code>	<code>10</code>	WAL
<code>max_replication_slots</code>	<code>10</code>	
<code>listen_addresses</code>	<code>*</code>	

`pg_hba.conf` IP

1. `hss` `prod_master_peers.yml` `connected_sites.hss`

2. 删除 OmniHSS 节点

删除节点

- 删除 PostgreSQL 的 OmniHSS
- 删除节点
- 删除节点
- 删除节点
- 删除节点
- 删除节点 `pg_hba.conf` 文件
- 删除节点

删除节点

```
ansible-playbook -i hosts site.yml --tags hss-remove-peer \
-e "peer_name=site_b confirm_remove=yes"
```

`confirm_remove=yes` 删除节点

删除节点

- 删除节点
- 删除 `pg_hba.conf` 文件 IP
- 删除节点
- 删除节点

删除节点

删除节点

```
ansible-playbook -i hosts site.yml --tags hss-add-peer \
-e "peer_name=site_b peer_host=10.80.20.140"
```



Prometheus

postgres_exporter は PostgreSQL のメトリクスを Prometheus にエクスポートするツールです。9187 番ポートで Prometheus に接続します。

インストール

メトリクス名	タイプ	説明
pg_replication_slots_active	Gauge	現在アクティブなレプリケーションスロットの数。1以上0以下。
pg_replication_slots_pg_wal_lsn_diff	Gauge	レプリケーションスロットの WAL レジスタの差分。レプリケーションスロットの WAL レジスタの差分。
pg_stat_replication_pg_current_wal_lsn_bytes	Gauge	レプリケーションスロットの WAL レジスタの現在のバイト数。

Prometheus

```
# レプリケーションスロットのアクティブ数を監視
pg_replication_slots_active{slot_name=~"sub_from_.*"}

# レプリケーションスロットの WAL レジスタの差分を監視
pg_replication_slots_pg_wal_lsn_diff / 1048576

# レプリケーションスロットのアクティブ数を監視
pg_replication_slots_active == 0
```

インストール

項目	内容	単位	単位
pg_replication_slots_active	<code>pg_replication_slots_active == 0</code> 時 60 分	分	pg_replication_slots_active WAL 削除
WAL 削除	<code>pg_replication_slots_pg_wal_lsn_diff > 104857600</code> 時 5 分	分	pg_replication_slots_pg_wal_lsn_diff 100MB 削除 pg_replication_slots_pg_wal_lsn_diff
WAL 削除	WAL 削除 <code>lost</code>	分	pg_replication_slots_pg_wal_lsn_diff pg_replication_slots_pg_wal_lsn_diff

Ansible 実行

OmniHSS 実行環境構築

1. 実行環境構築
2. 実行環境構築
3. WAL 削除 100MB
4. WAL 削除 `reserved` 時 `extended` 時 `lost`

実行環境構築

確認

```
# 確認pg_stat_subscription
sudo -u postgres psql -d omnihss -c \
    "SELECT subname, pid IS NOT NULL as worker_alive FROM
pg_stat_subscription;"

# 確認pg_replication_slots
sudo -u postgres psql -d omnihss -c \
    "SELECT slot_name, active,
        pg_wal_lsn_diff(pg_current_wal_lsn(), confirmed_flush_lsn) as
lag_bytes,
        wal_status
    FROM pg_replication_slots WHERE slot_type = 'logical';"

# 確認pg_subscription
sudo -u postgres psql -d omnihss -c "SELECT count(*) FROM
subscriber;"
```

確認

確認pg_stat_subscription

pg_stat_subscription 確認pg_stat_subscription pid IS NULL Ansible 確認pg_stat_subscription

確認pg_replication_slots

- 確認pg_replication_slots DDL 確認
- 確認pg_replication_slots
- 確認pg_replication_slots WAL 確認

確認pg_subscription

1. 確認 PostgreSQL 確認pg_subscription

```
journalctl -u postgresql | grep "logical replication" | tail
-20
```


- DDL
- DDL
-

```
ALTER SUBSCRIPTION sub_from_<peer> ENABLE;
```

Configuration

`pg_replication_slots` `active = false` Prometheus

-
- PostgreSQL
- IP 5432

-

```
pg_isready -h <peer_ip> -U <db_user> -d omnihss
```

- PostgreSQL
-
- IP 5432

WAL

`pg_replication_slots` `wal_status = 'lost'` Ansible "WAL"

WAL `max_slot_wal_keep_size`

-

```
DROP SUBSCRIPTION sub_from_<peer>;
```

2. 削除スロットを削除

```
SELECT pg_drop_replication_slot('sub_from_<this_site>');
```

3. OmniHSS のインストール時に `copy_data = true` を指定してインストール

WAL 監視

pg_replication_slots_pg_wal_lsn_diff を Prometheus に監視

監視項目

- 監視対象データベース
- 監視対象データベースのレプリケーション
- 監視対象データベースの WAL

監視方法

1. 監視対象データベース

```
SELECT slot_name,  
       pg_wal_lsn_diff(pg_current_wal_lsn(), confirmed_flush_lsn) as  
lag_bytes  
FROM pg_replication_slots WHERE slot_type = 'logical';
```

2. 監視対象データベースのレプリケーション

3. 監視対象データベースの WAL

4. 監視対象データベースのレプリケーションの WAL

インストール

インストール時に OmniHSS のインストール

1. 監視対象データベースのインストール/アップグレード

2. Ansible を利用してインストール/アップグレード

3. 監視対象データベースのインストール

```
/opt/omnihss/bin/hss rpc "Hss.Command.Database.migrate()"
```


OmniHSS

←

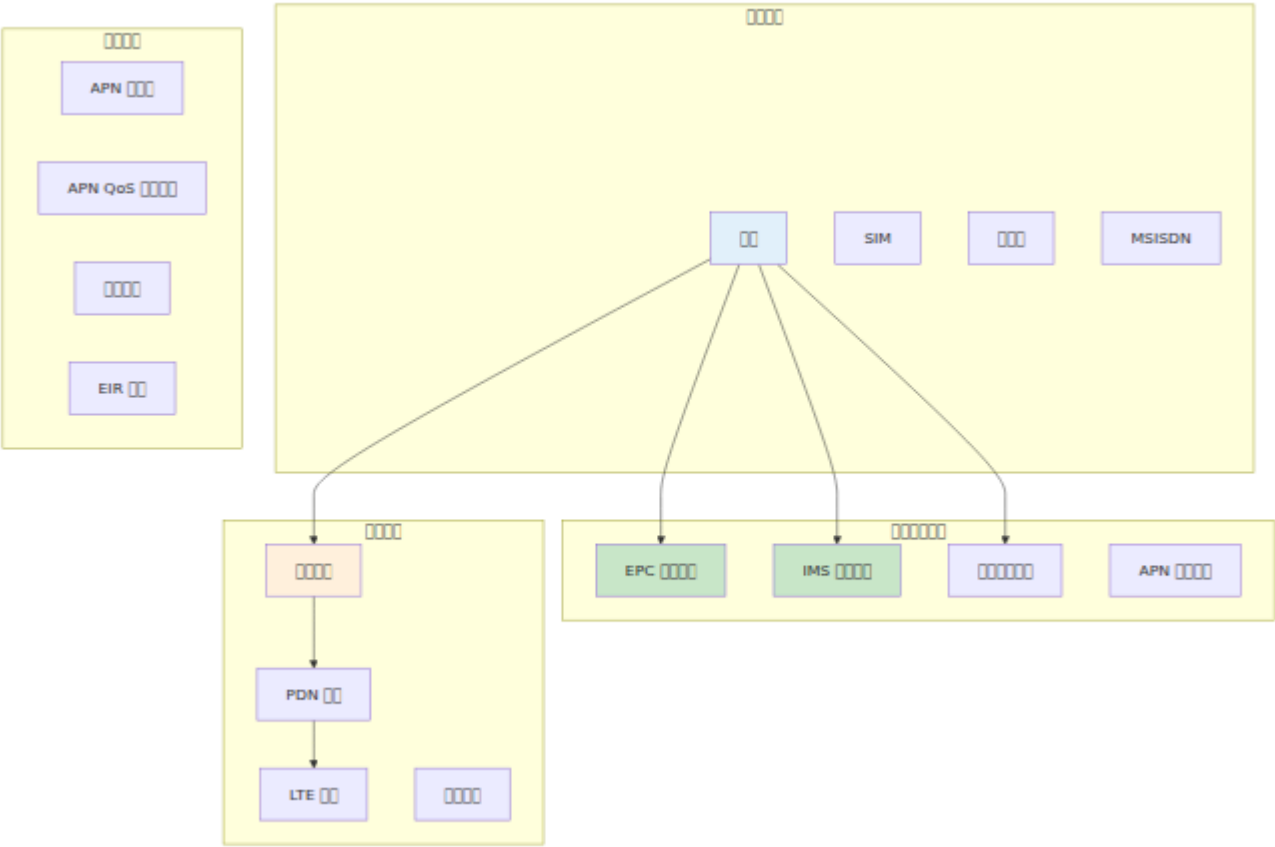
11

- 
- 
- 
- 
- 
- 
- 
- 

--	--	--	--

OmniHSS ██████████

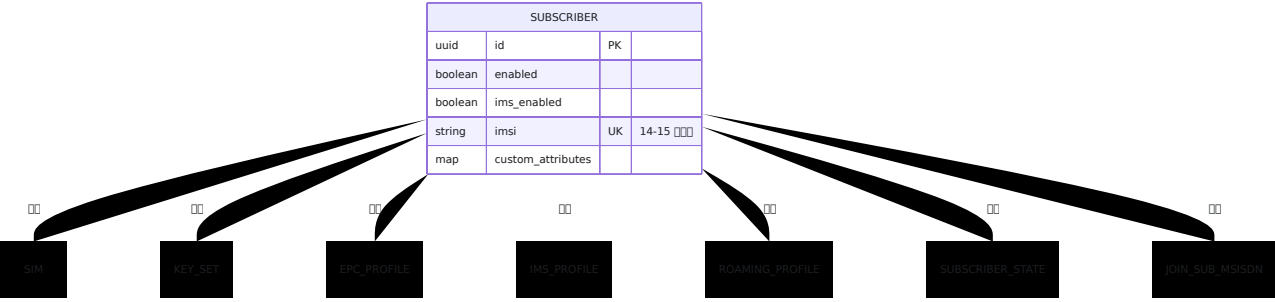
□□□□



□□□□

□□

□□□□□□□□□□□□□□



□□:

項目	項目
IMS 設定	key_set 及び sim
LTE 設定 S6a 設定	epc_profile
IMS / Cx 設定	ims_profile 及び MSISDN
その他	roaming_profile

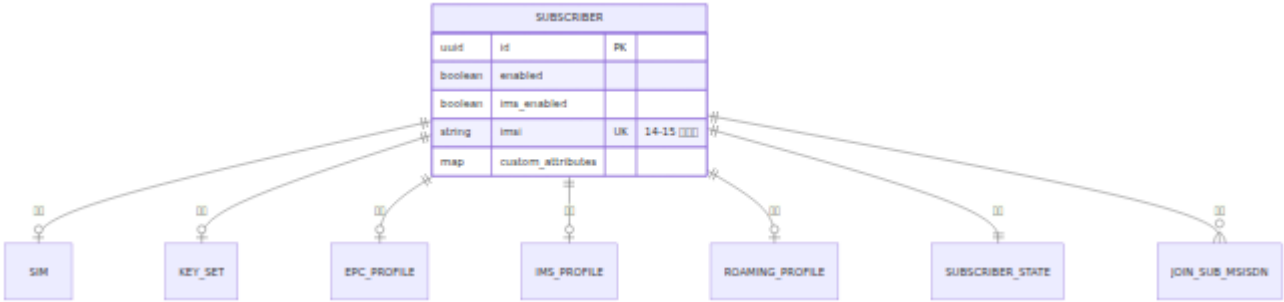
IMS 設定は HSS 設定と関連付けられています。

設定:

- IMS 設定
- IMS 14-15 設定
- MSISDN 設定
- 設定
- enabled 設定 IMS
- ims_enabled 設定 IMS
- IMS 設定 IMS 設定 IMS 設定 IMS 設定 IMS

SIM

SIM 設定 SIM 設定



設定:

欄名	型別	説明	単位
iccid	string	ICCID (Integrated Circuit Card Identifier) ID	文字列
sim_vendor	string	SIM 製造元	文字列
batch_name	string	バッチ名	文字列
is_esim	boolean	eSIM かどうか	ブール値
pin1, pin2	string	PIN 番号	文字列
puk1, puk2	string	PUK 番号	文字列
adm1 - adm10	string	ADM 番号	文字列
kic, kid	binary	OTA キー	バイナリ

注:

- ICCID は SIM カードの一意の識別子
- SIM 製造元は SIM カードの製造元を示す
- PIN/PUK は SIM カードのセキュリティ番号
- ADM は SIM カードの管理者番号
- KIC/KID は SIM カードの OTA キー

備考

この情報は参考情報です。

KEY_SET			
uuid	id	PK	
binary	ki		128 ㄱ
binary	opc		128 ㄱ
binary	op		128 ㄱ
binary	amf		16 ㄱ
bigint	sqn		48 ㄱㄱㄱ
string	authentication_algorithm		

ㄱㄱ



ㄱㄱ:

項目	型別	説明	長さ
ki	binary	秘密鍵	128 ビット16 バイト
opc	binary	オペレーションコード	128 ビット
op	binary	オペレーションコード OPC	128 ビット
amf	binary	認証モード	16 ビット2 バイト
sqn	bigint	シーケンス番号	48 ビット
authentication_algorithm	string	認証アルゴリズム	文字列 "milenaage"
ota_counter	bigint	OTA カウンタ	48 ビット

項目:

- 秘密鍵
- Ki 秘密 SIM 秘密鍵
- オペレーションコード OPC オペレーションコード OP
- SEQN シーケンス番号
- Milenaage 認証アルゴリズム

項目:

SIM (sim_id) (key_set_id) SIM SIM
 SIM

MSISDN

MSISDN 電話番号

MSISDN			
uuid	id	PK	
string	msisdn	UK	1-15 数字

参照

JOIN_SUB_MSISDN		
uuid	subscriber_id	FK
uuid	msisdn_id	FK

参照



例:

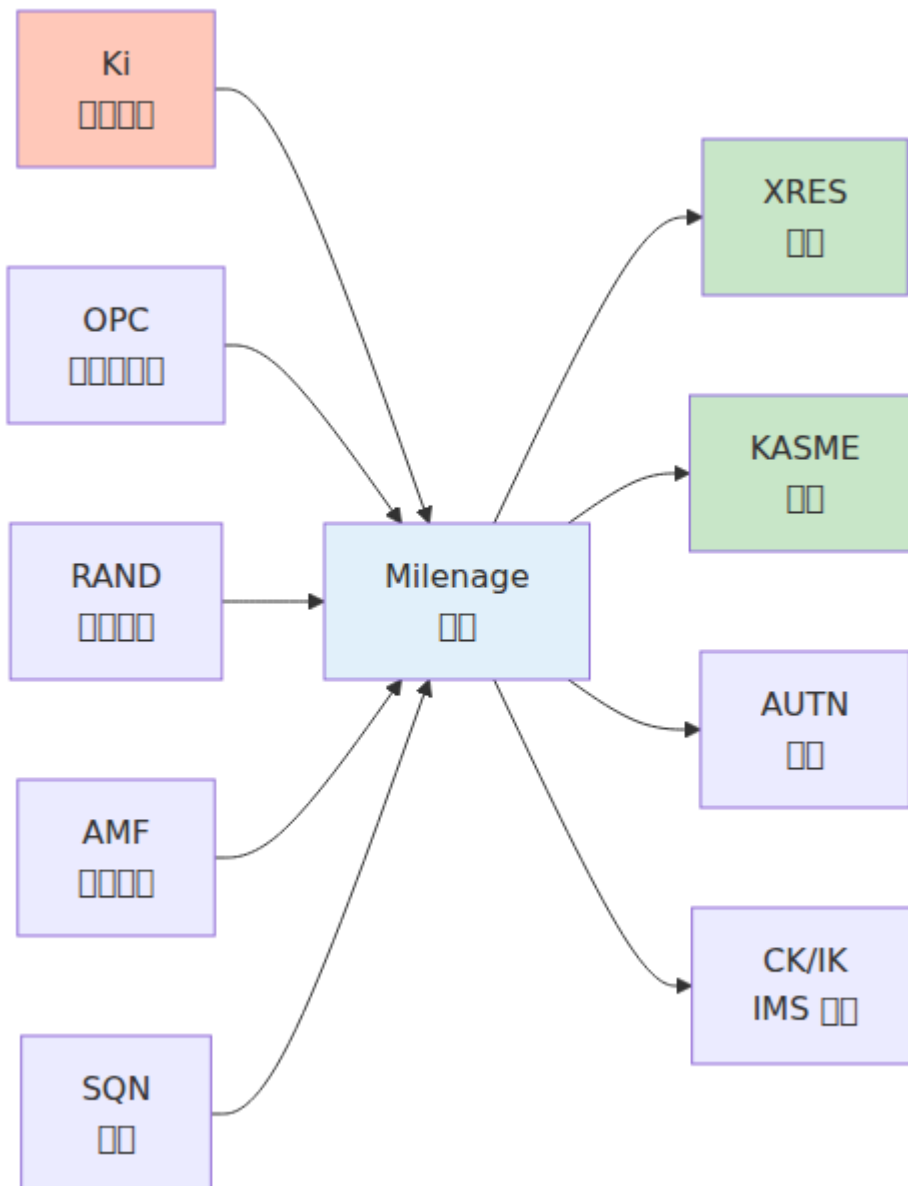
属性	型	制約	注
msisdn	string	ユニーク ISDN 番号	1-15 数字E.164 番号

備考:

- MSISDN 電話番号
- 電話番号 MSISDN番号 IMSI番号

- MSISDN IMSI
- MSISDN API GET /subscriber/msisdn/:msisdn Diameter IMSI ICCID
- + - + 14155551234 +14155551234
- Diameter Cx/Rx + S-CSCF P-CSCF + SIP URI

MSISDN



□□□□□□

EPC □□□□

EPC □□□□□□ LTE □□□□□□□□

EPC_PROFILE			
uuid	id	PK	
string	name	UK	
integer	ue_ambr_dl_kbps		
integer	ue_ambr_ul_kbps		
string	network_access_mode		□□□□□□□□
integer	tracking_area_update_interval_seconds		



JOIN_EPC_APN



SUBSCRIBER

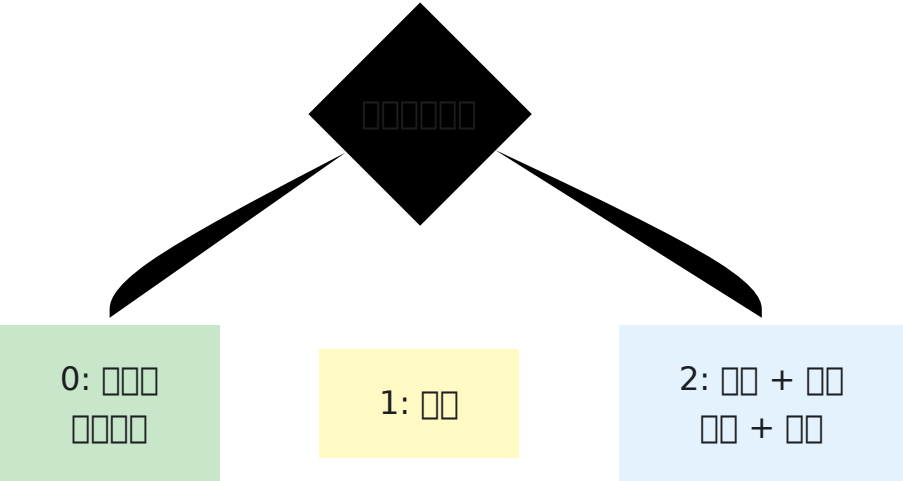
□□

APN_PROFILE

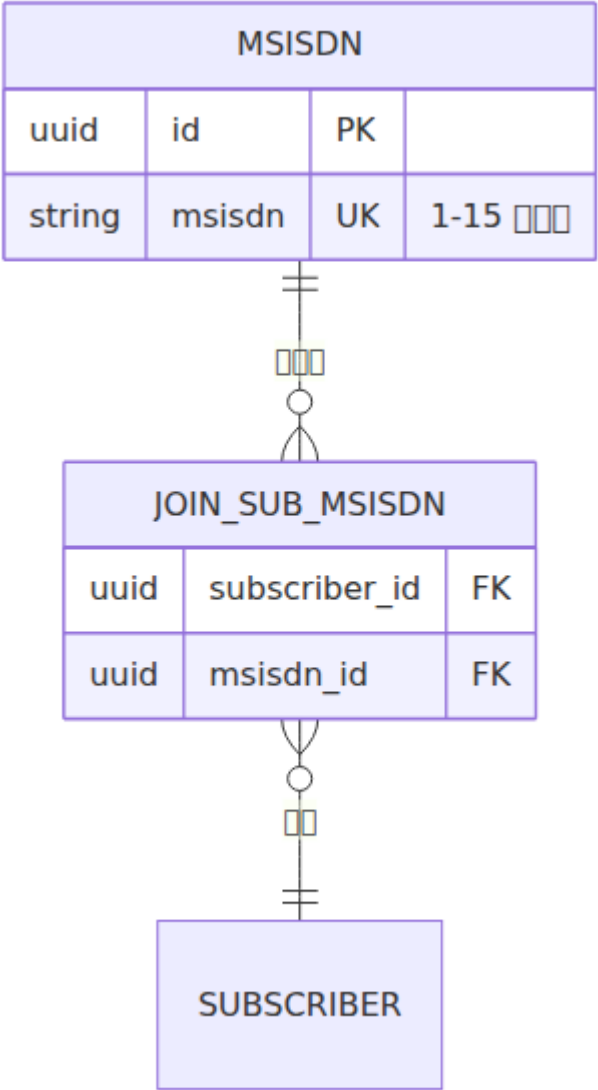
□□:

項目	型	範囲	単位
name	string	0~31 文字	
ue_ambr_dl_kbps	integer	0~32767 整数	Kbps
ue_ambr_ul_kbps	integer	0~32767 整数	Kbps
network_access_mode	string	0~31 文字	"0" または "1" 文字
tracking_area_update_interval_seconds	integer	TAU 区間 整数	

AMBRの範囲:



AMBRの範囲:



IMS 0000

IMS 00000000/00000000

IMS_PROFILE		
uuid	id	PK
string	name	UK
text	ifc_template	

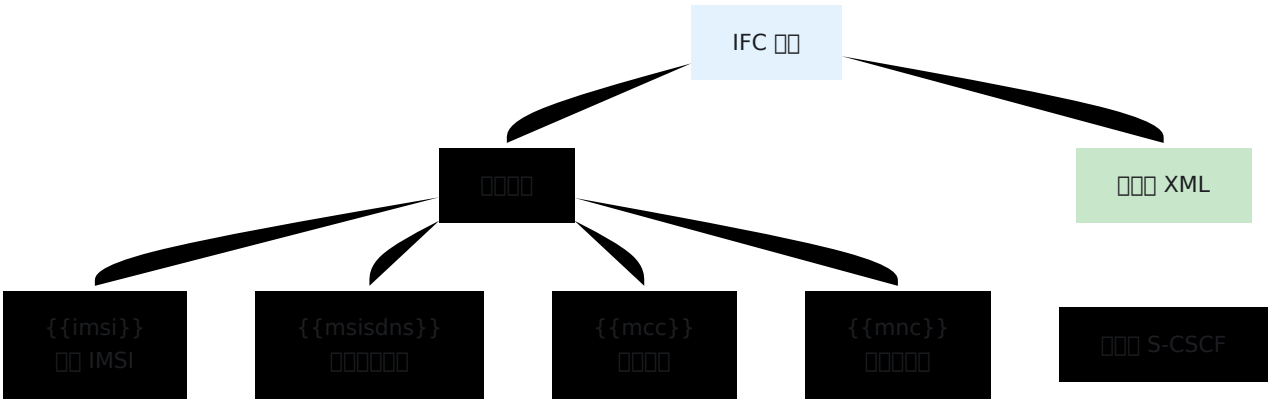
数据



数据:

名称	类型	描述	备注
name	string	用户名称	必填
ifc_template	text	用户模板 XML 内容	必填 XML

IFC 模板:



数据:

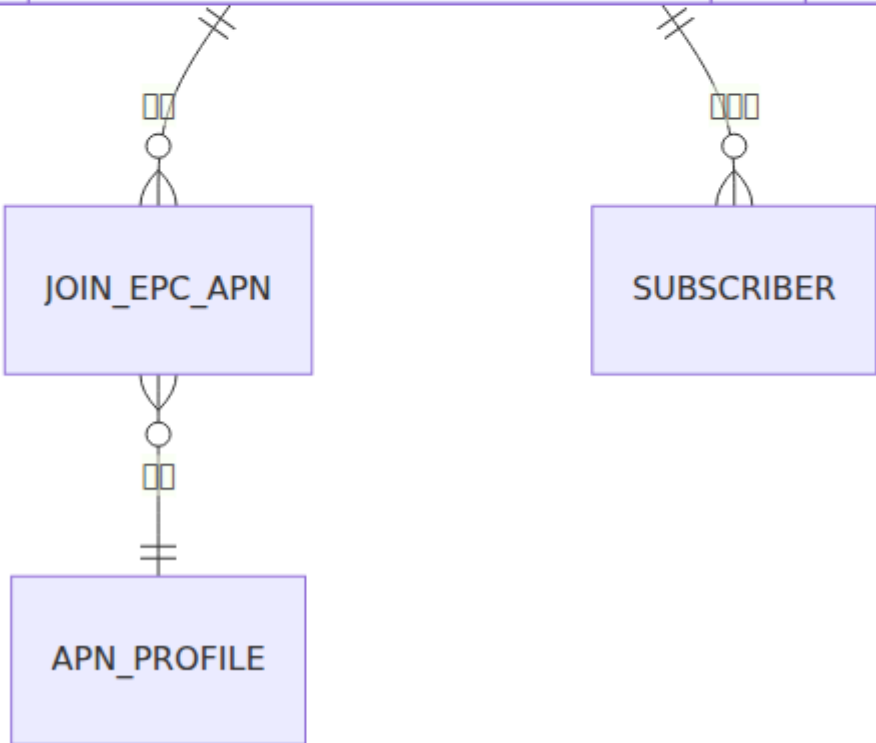
- IFC模板用于生成 IMS 数据
- 模板数据

- IMS S-CSCF

APN

APN

EPC_PROFILE			
uuid	id	PK	
string	name	UK	
integer	ue_ambr_dl_kbps		
integer	ue_ambr_ul_kbps		
string	network_access_mode		00000000
integer	tracking_area_update_interval_seconds		



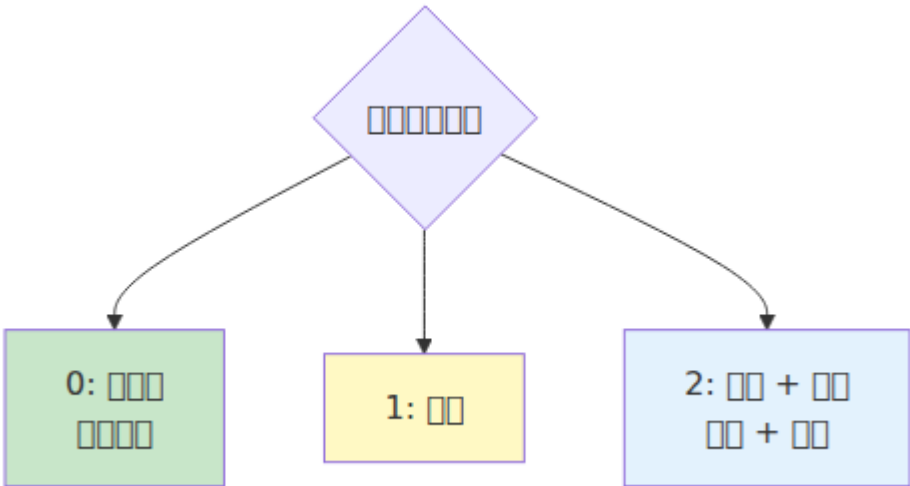
□□□□ ■
□□□□ ■

APN

APN_IDENTIFIER			
uuid	id	PK	
string	apn	UK	
string	ip_version		ipv4, ipv6, ipv4v6, 0 ipv4_or_ipv6

Field	Type	Constraints	Values
apn	string	APN 00	"internet", "ims", "mms"
ip_version	string	IP 0000	0000

IP 0000:



APN QoS 0000

APN_QOS_PROFILE		
uuid	id	PK
string	name	UK
integer	qci	
integer	allocation_retention_priority	
integer	apn_ambr_dl_kbps	
integer	apn_ambr_ul_kbps	
boolean	pre_emption_capability	
boolean	pre_emption_vulnerability	

QoS :

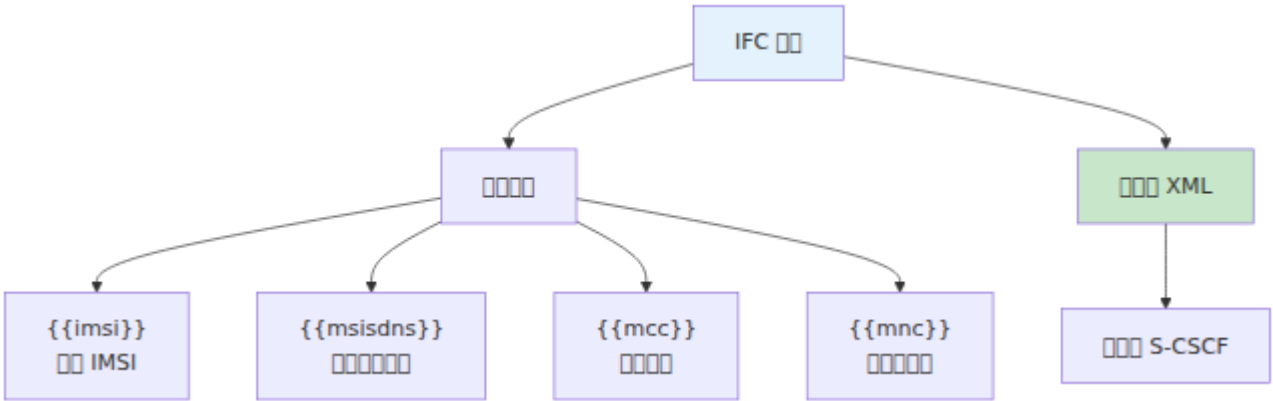
欄名	説明	範囲	デフォルト値
qci	QoS プロファイル	1-9	QCI 9
allocation_retention_priority	ARP 値	1-15	8
apn_ambr_dl_kbps	APN 最大ダウンロード速度	0+	0
apn_ambr_ul_kbps	APN 最大アップロード速度	0+	0
pre_emption_capability	優先権上乗せ能力	true/false	false
pre_emption_vulnerability	優先権上乗せ脆弱性	true/false	true

QCI :

□□□□:

ROAMING_RULE			
uuid	id	PK	
string	name	UK	
string	mcc		3 位
string	mnc		2-3 位
string	data_action		位
string	ims_action		位

：



位

位

位

SUBSCRIBER_STATE		
uuid	id	PK
string	last_seen_mme	
string	last_seen_realm	
integer	last_seen_rat_type	
datetime	last_seen_at	
string	last_seen_mcc	
string	last_seen_mnc	
string	last_seen_tac	
string	last_seen_cell_id	
string	last_seen_enodeb_id	
string	last_seen_eci	
string	assigned_scscf	
string	ims_public_identity	
text	sh_repository_data	

SUBSCRIBER

PDN_SESSION

CIRCUIT_SESSION

属性:

属性:

- last_seen_mcc, last_seen_mnc - 国家コード
- last_seen_tac - TAC
- last_seen_cell_id - 小区ID
- last_seen_enodeb_id - eNodeB ID
- last_seen_eci - E-UTRAN 国家コード

IMS:

- `last_seen_mme` - IMS MME
- `last_seen_realm` - MME Diameter realm
- `last_seen_rat_type` - IMS RAT type (LTE, 5G, etc.)

IMS:

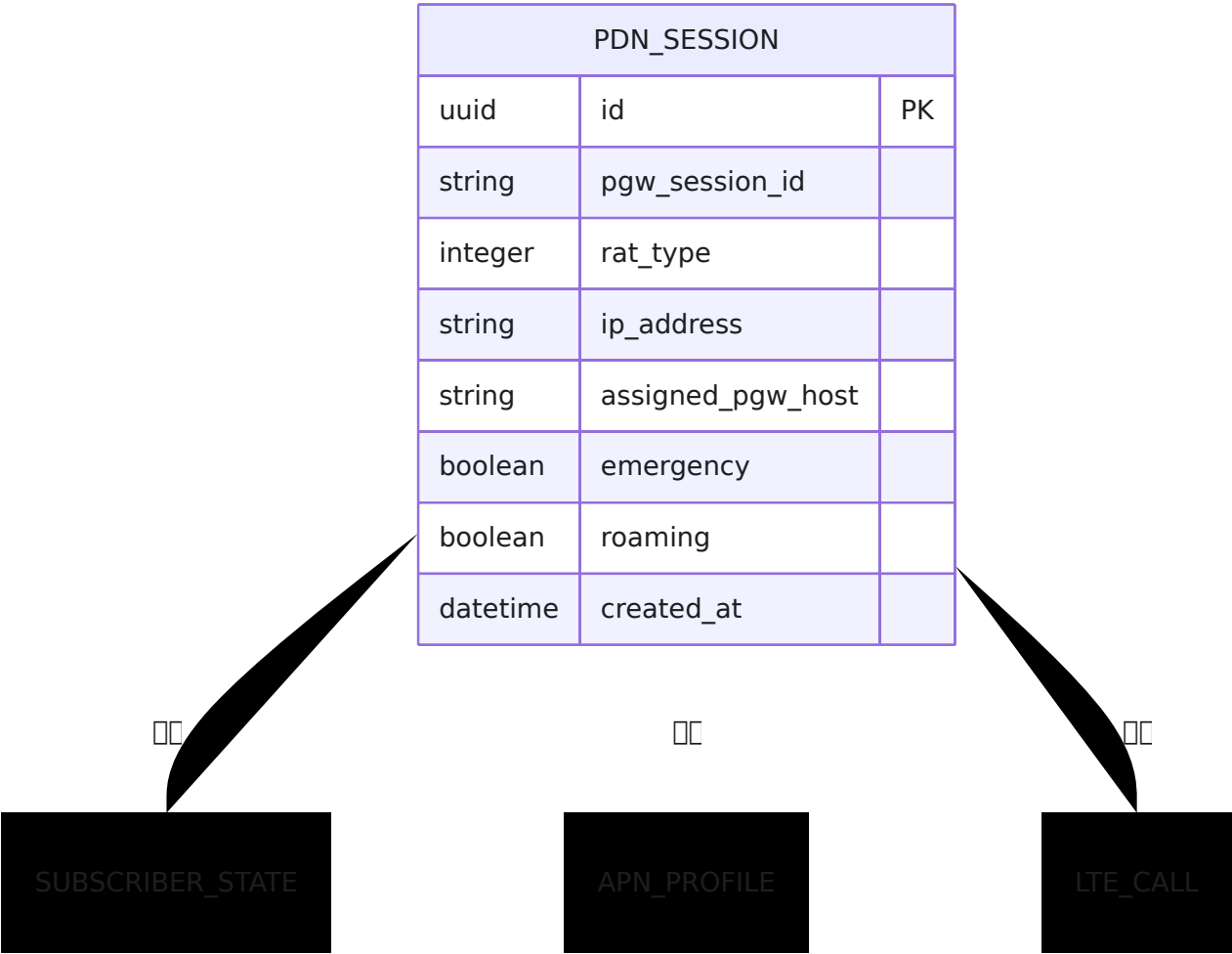
- `assigned_scscf` - Assigned S-CSCF
- `ims_public_identity` - SIP URI
sip:+14155551234@ims.example.com
- `sh_repository_data` - IMS service data

IMS:

- `last_seen_at` - Last seen at Diameter realm
- `last_*_at` - Last seen at various realms

PDN

PDN (Public Data Network)



PDN

APN_IDENTIFIER			
uuid	id	PK	
string	apn	UK	
string	ip_version		ipv4, ipv6, ipv4v6, [] ipv4_or_ipv6

LTE

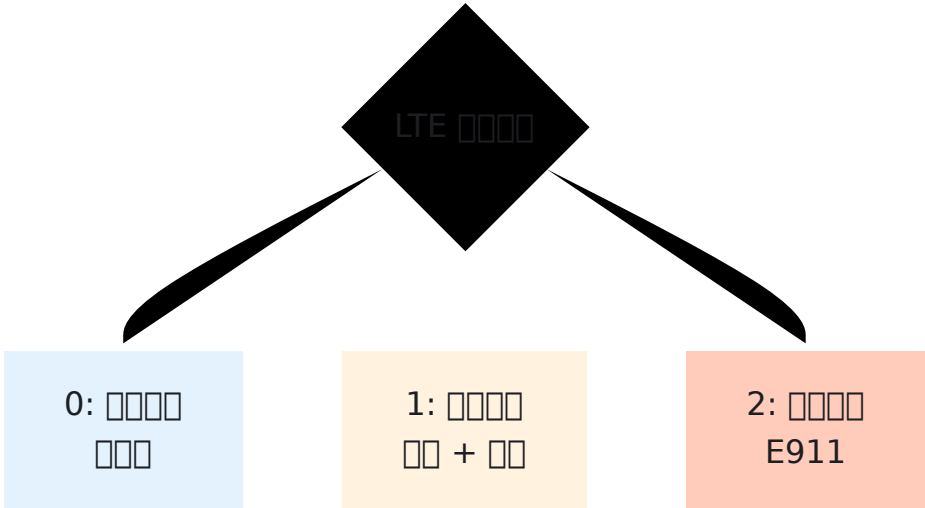
LTE VoLTE

LTE_CALL		
uuid	id	PK
string	pcscf_session_id	
string	assigned_pcscf_host	
string	codec	
integer	type	
boolean	dedicated_bearer_active	
integer	dedicated_bandwidth_ul	
integer	dedicated_bandwidth_dl	
text	flow_rules	
datetime	created_at	

□□

PDN_SESSION

□□□□□

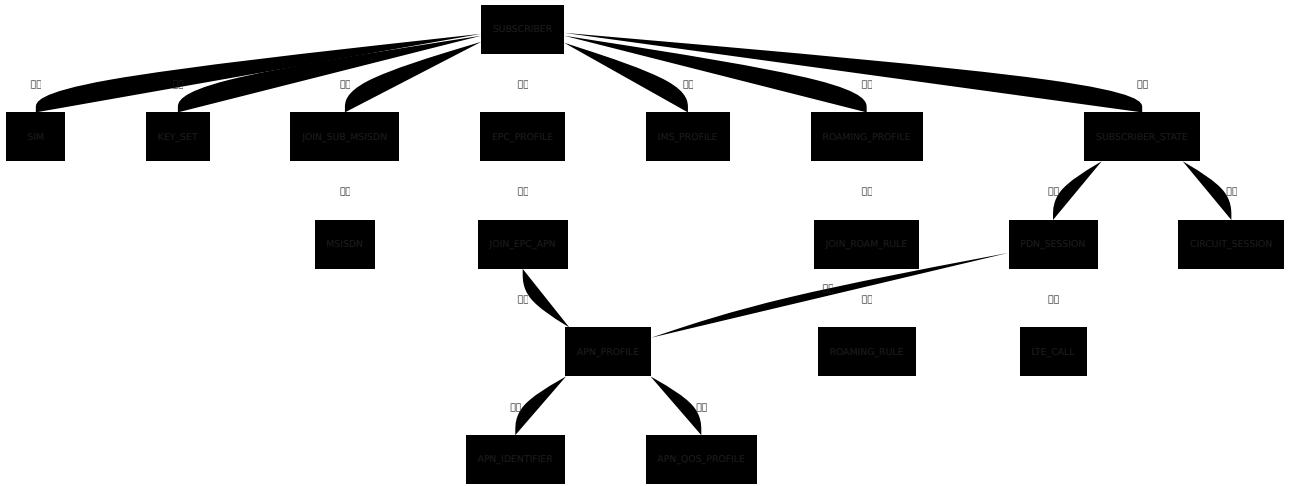


VoLTE 00000

APN_QOS_PROFILE		
uuid	id	PK
string	name	UK
integer	qci	
integer	allocation_retention_priority	
integer	apn_ambr_dl_kbps	
integer	apn_ambr_ul_kbps	
boolean	pre_emption_capability	
boolean	pre_emption_vulnerability	

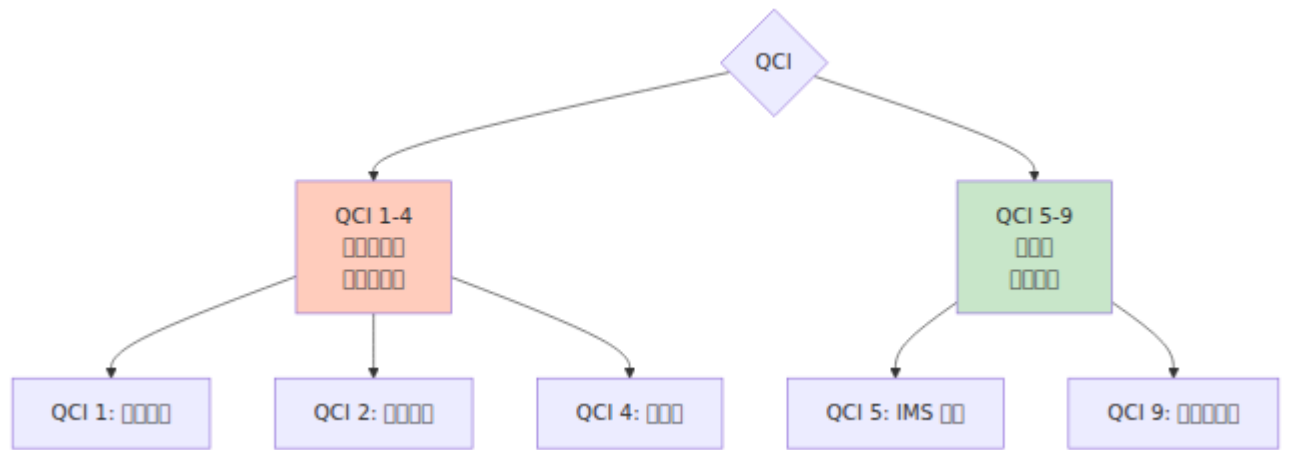
□□□□□

□□□□□□

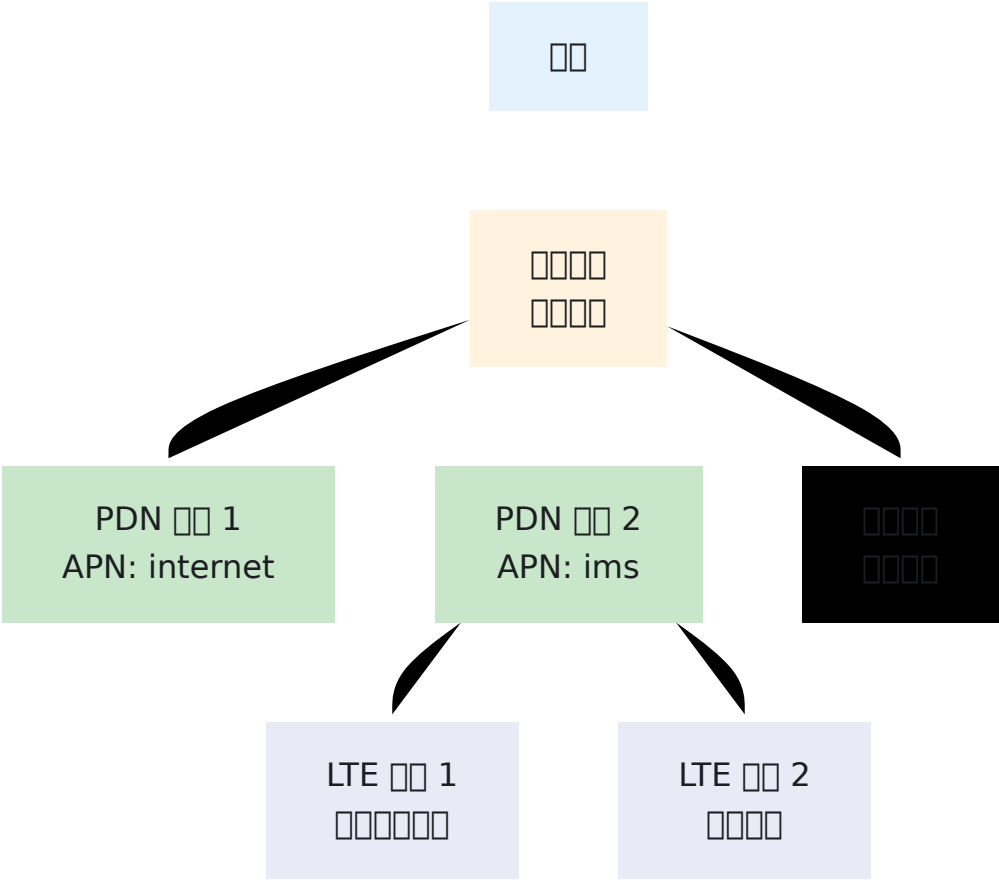


□□□□□

□□□□□□□□□□□□□□□□□□



□□□□□□



□□□□□□

□□□□□□□□



□□□□□□□□

Create_Prerequisites

□□□□□□□□

□□□□□□
□□ EPC □□□□□□ APN□
□□ IMS □□□□□□□□
□□□□□□□□□□□□
□□ SIM□□□□

Create_Subscriber

enabled=false

enabled=true

Disabled

□□ enabled=true □□ enabled=false

Enabled

□□□□

□□□□□□□□

□□□□□□□□

□□□□

Active



IMS □□ IMS □□

IMS_Registered

VoLTE □□□□ □□□□

In_Call

□□□□□□

ROAMING_RULE			
uuid	id	PK	
string	name	UK	
string	mcc		3 □□□
string	mnc		2-3 □□□
string	data_action		□□□□□
string	ims_action		□□□□□

□□□□□

□□□□

S6a AIR □□

□□ IMSI □□□□

□□□□□

□□□□□□

□□□□□□
□□ Milenage

S6a AIA □□

□□□□□□

S6a ULR □□

□□ IMSI □□□□

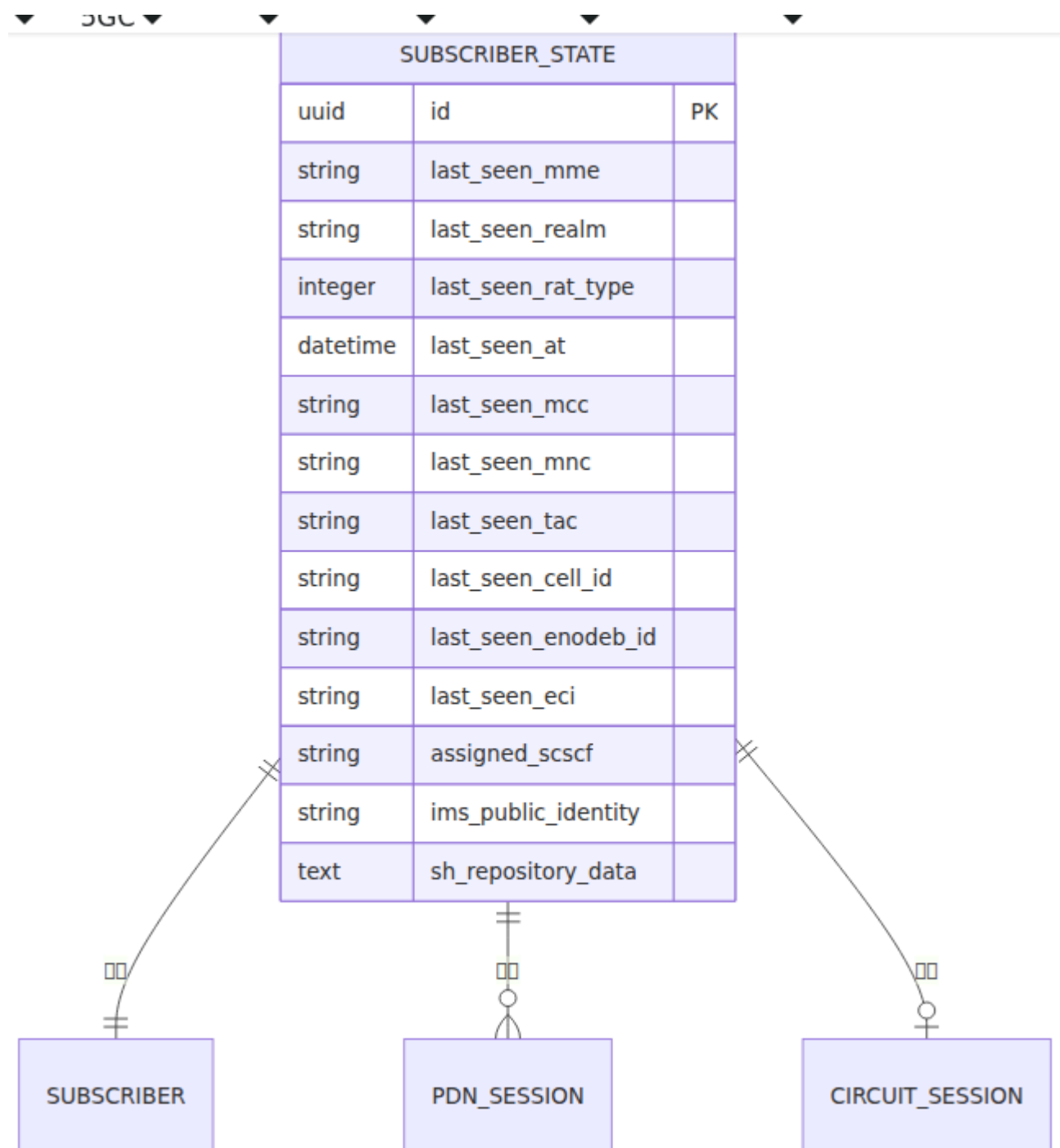
□□ EPC □□□□
+ APN □□□□

□□□□□□
□□□MME □

□□□□□□
AMBR□APN□QoS

S6a ULA □□

IMS



□□□□□□

Gx CCR-I □□

□□ IMSI □□□□

□□ APN □□□□
+ QoS □□□□

□□□□□□

□□ PDN □□
□□□□□□

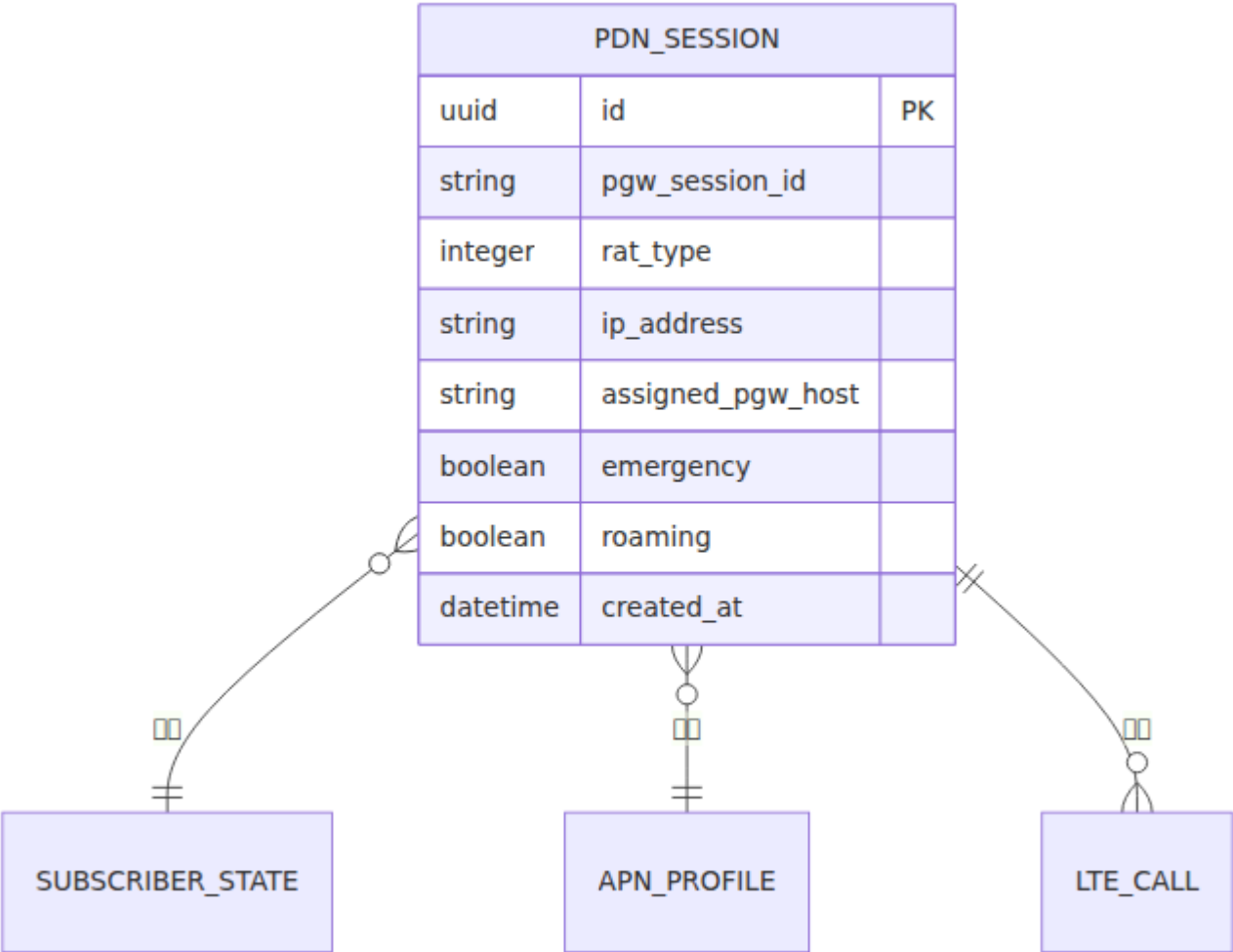
□□ QoS □□
AMBR□QCI□ARP

Gx CCA-I □□

□□□□□□

OmniHSS □□□□□□□□□□□□□□□□□□□□

□□□□□□□□



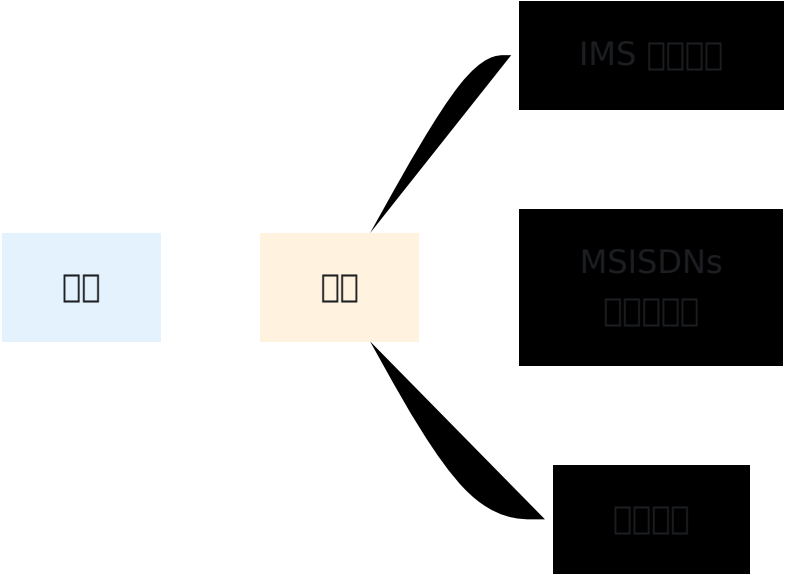
□□ S6a AIR - □□□□□□□□□□

□□□□□□□□□□



3GPP S6a ULR - 3GPP EPC 3GPP

3GPP IMS 3GPP



3GPP Cx SAR - 3GPP IMS 3GPP

← 3GPP | 3GPP API 3GPP →

Diameter

←

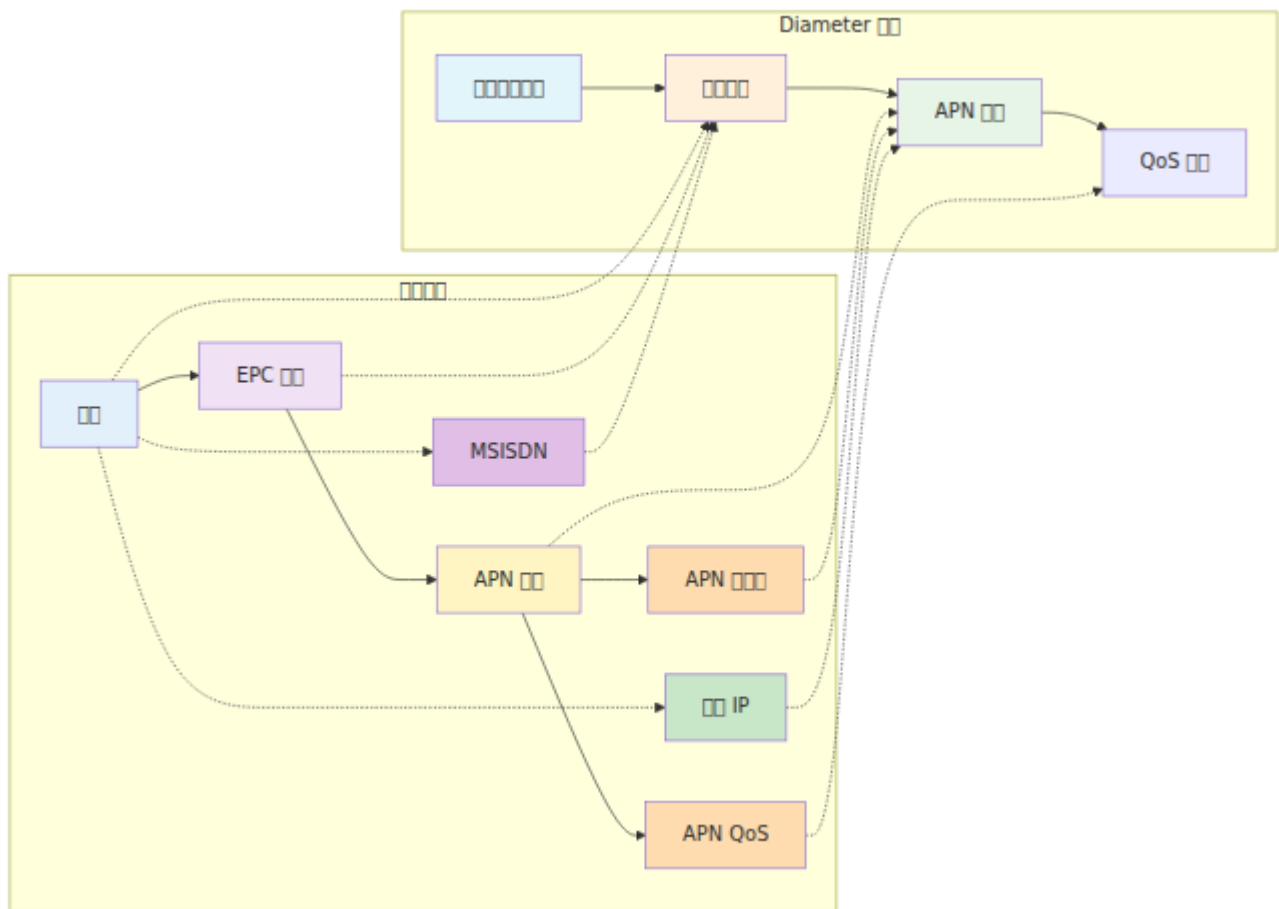
mermaid Diameter OmniHSS

- (S6a ULA)
- (S6a AIA)
- (Cx SAA)
- (Gx CCA)
- (Sh UDA)
- ME (S13 ECA)

(S6a ULA)

HSS LTE MME Diameter AVP

□□□□□



□□□□□□

□□□□	□□	□
subscriber.enabled	true/false	Sub Sta
msisdn.msisdn	'14155551234'	MS
epc_profile.ue_ambr_ul_kbps	50000	Ma Re Ba UL
epc_profile.ue_ambr_dl_kbps	100000	Ma Re Ba DL
epc_profile.network_access_mode	'packet_only'	Ne Ac Mo
apn_identifier.apn	'internet'	Se Se
apn_identifier.ip_version	'ipv4v6'	PD
apn_qos_profile.qci	9	Qc Id
apn_qos_profile.allocation_retention_priority	8	Pr Le

項目名	値	説明
apn_qos_profile.pre_emption_capability	false	Pre-emption Capability
apn_qos_profile.pre_emption_vulnerability	true	Pre-emption Vulnerability
apn_qos_profile.apn_ambr_ul_kbps	25000	APN UL AMBR
apn_qos_profile.apn_ambr_dl_kbps	50000	APN DL AMBR
static_ip.ipv4_static_ip	'100.64.1.1'	Static IPv4 Address (If enabled)
static_ip.ipv6_static_ip	'2606:4700::1111'	Static IPv6 Address (If enabled)

注意事項

- AMBR** 値は 10000 kbps から 100000 kbps まで指定可能
- IP** 値は 0=IPv4, 1=IPv6, 2=IPv4v6, 3=IPv4_or_IPv6
- enabled**: true → 0 (SERVICE_GRANTED), false → 1 (OPERATOR_DETERMINED_BARRING)
- APN 値は 0, 1, 2...
- IP** 値は static_ips 値を参照

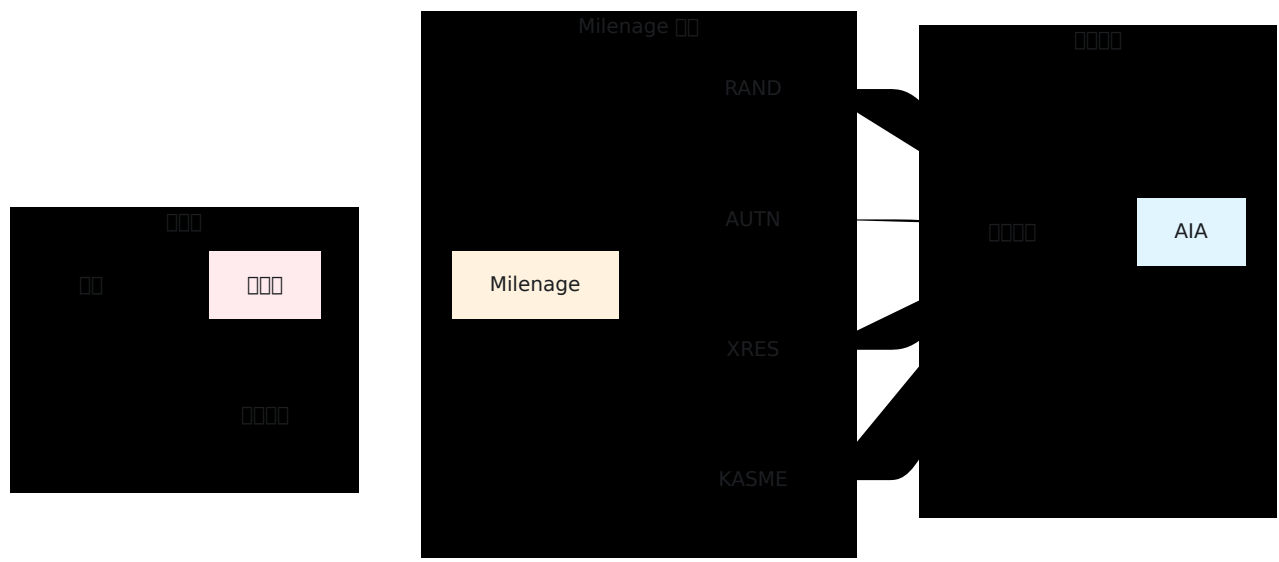
関連項目

- どのPLMNに `roaming_profile.roaming_rules`
- `subscriber.enabled == true`
- APNがIMSのIMS APN

認証 (S6a AIA)

LTE/EPC 認証

概要



手順

1. 認証に必要なパラメータ `key_set` を取得
2. **SQL** を実行して必要なデータを取得
3. **Milenage** アルゴリズム (3GPP TS 35.206) を実行
4. **KASME** アルゴリズム (3GPP TS 33.401) を実行

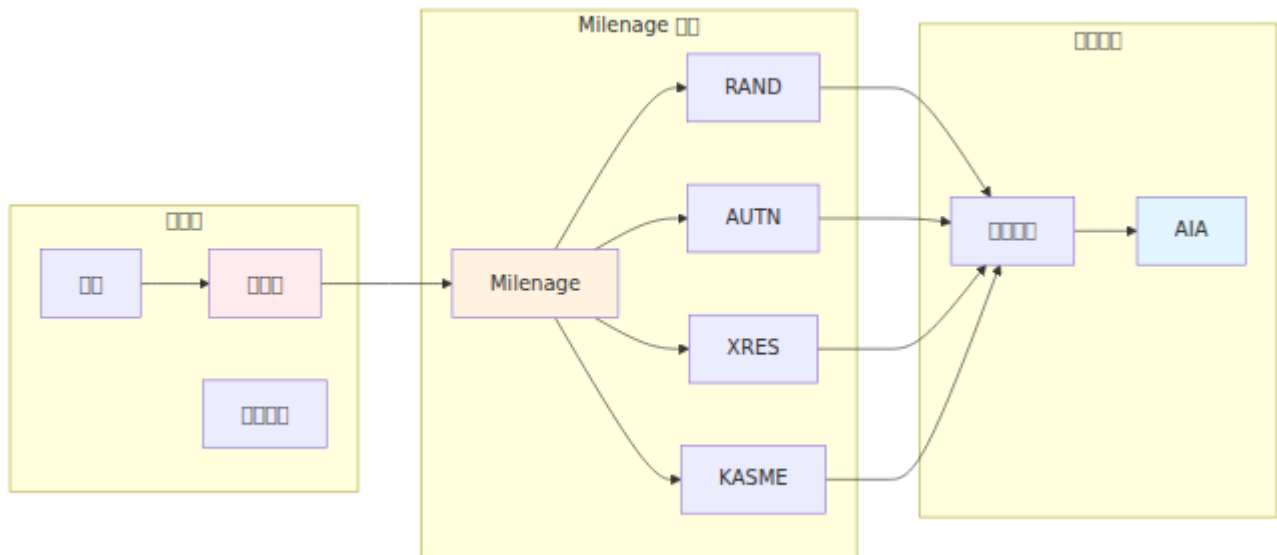
結果

- SQLの結果
- Ki/OPc を取得して HSSに送信
- AUTN を生成して SQLに AMF を追加
- Milenage アルゴリズムで UE 認証

AAA (Cx SAA)

AAA HSS IMS AAA S-CSCF

AAA



AAA

1. **IFC** AAA XML AAA `ims_profile.ifc_template`
2. AAA AAA AAA `{{msisdn}}` AAA `{{imsi}}` AAA `{{impu}}`
3. **S-CSCF** AAA `subscriber_state.assigned_scscf` AAA S-CSCF
4. **IMS** AAA AAA `sip:++{msisdn}@{ims_domain}` AAA `tel:++{msisdn}`

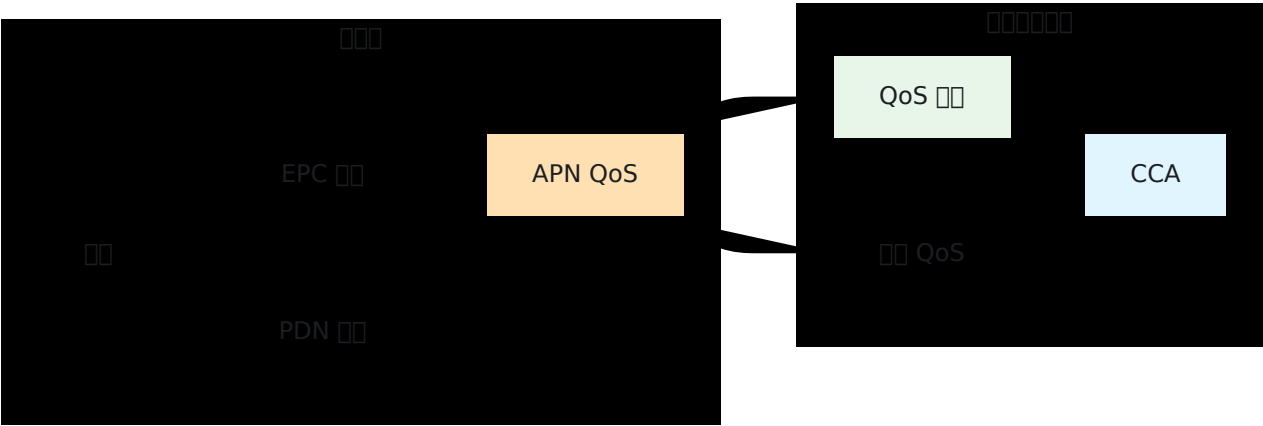
IFC AAA

- `{{msisdn}}` - AAA MSISDN
- `{{imsi}}` - AAA IMSI
- `{{impu}}` - IMS AAA `subscriber_state`
- `{{impi}}` - IMS AAA `IMSI@realm`

AAA (Gx CCA)

AAA PCRF AAA PGW

概要



動作

1. `pdn_session` の作成
2. **QoS** の設定 (APN QoS の QCI の設定)
3. `CC-Request-Type` の設定 (INITIAL (1) / UPDATE (2) / TERMINATION (3))

動作の流れ

- `INITIAL_REQUEST` の送信 (PDN の作成)
- `UPDATE_REQUEST` の送信 (PDN の更新)
- `TERMINATION_REQUEST` の送信 (PDN の削除)

Sh UDA (Sh UDA)

HSS の Sh UDA AS (Sh UDA AS) の動作

概要



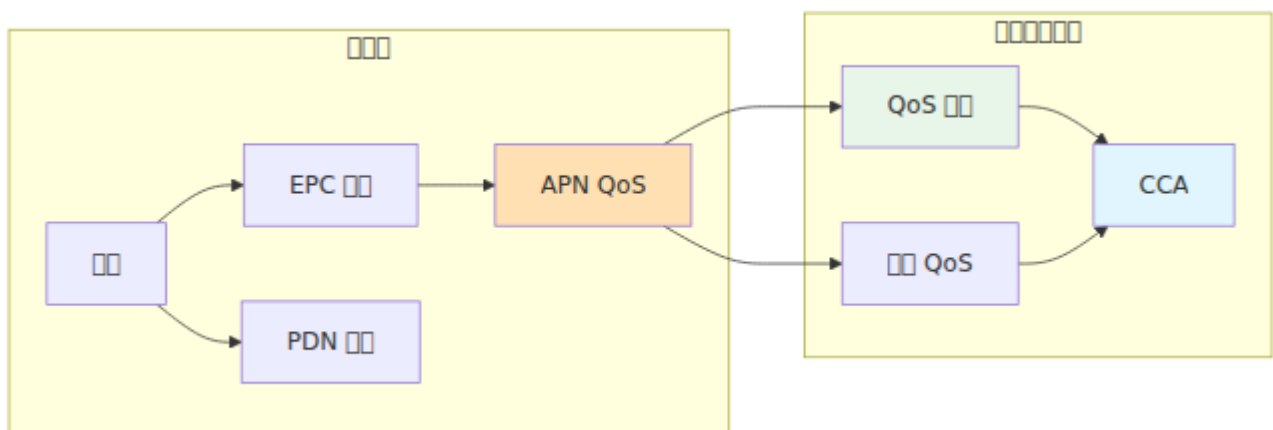
5/5

1. subscriber_state.sh_repository_data XML
- 2.
3. IMS
- 4.

ME (S13 ECA)

ME 00000000 EIR 000000 MME 00 IMEI 000

□ □ □ □ □



□□□□□

1. **IMEI** □□□□□□□□□□□□□□□□□□
2. □□ **TAC** □□□□□□□□□□□□□□□□ 8 □□□□
3. □□□□□□□□□□ IMEI□□□□□□□□
4. □□□□□□
 - 0 = □□□□□□□□□□
 - 1 = □□□□□□□/□□□
 - 2 = □□□□□□□□□□
 - 5 = □□□□□□□□□□

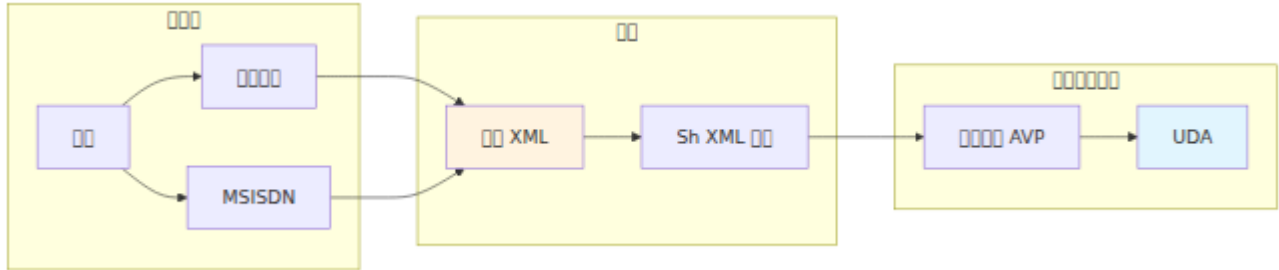
111

- IMEI
 - TAC
 -
 -
-

Diameter AVP

□□□□□

□□□□□□



□□□□

□□□□□□

□□□□□□ Diameter □□□□□□□□

- **S6a** - LTE/MME □□□□□□□□□□
- **Cx** - IMS/CSCF □□□□ IMS □□□□□□□□
- **Sh** - IMS/AS □□□□□□□□□□
- **Gx** - PCRF □□□□□□□□□□
- **Rx** - IMS/AF □□□□□□□□
- **S13** - EIR □□□□ IMEI □□
- **SWx** - WiFi/IMS □□□□❓❓❓ 3GPP □□□□

□□□□

□□□□□□□□□□□□□□

- □□ - □□□□□□□□□□ IMSI
- □□□ - □□□□□□□□□□
- **EPC** □□ - LTE □□□□
- **APN** □□ - □□□□□□
- **IMS** □□ - □□ IFC □□□□ IMS □□□□

- **IMEI** - 國際移動設備識別碼
- **ICCID** - 國際卡識別碼
- **PDN** - 網路數據網路
- **IP** - IP 位址
- **EIR** - IMEI 資料庫

← [API 說明](#) | [API 說明](#) → | [API 說明](#) →

OmniHSS 部署架构图

← 部署图

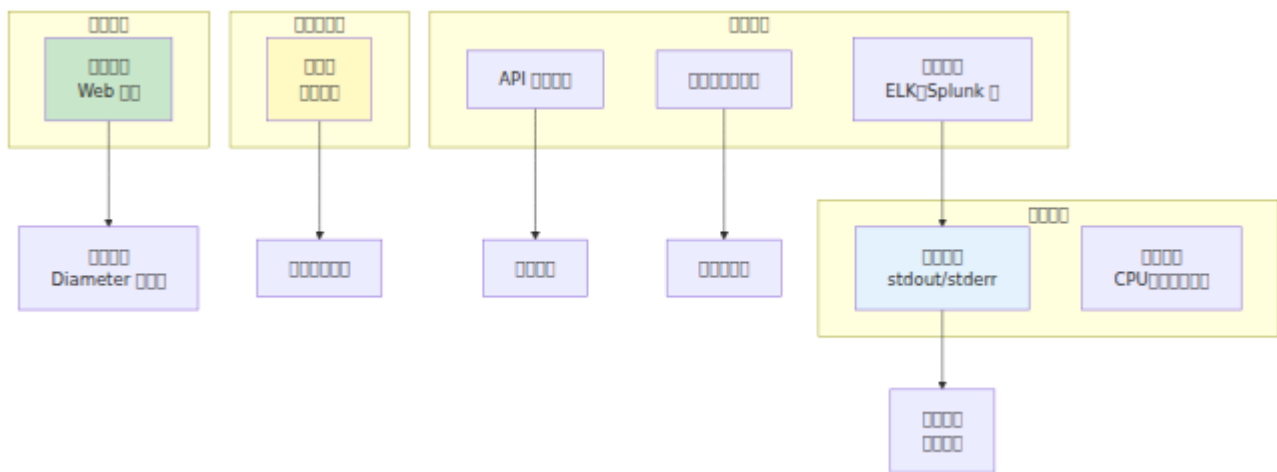
部署

- 部署
- 部署图
- 部署图
- 部署
- 部署图
- 部署图
- 部署

部署图

OmniHSS 部署图展示了 OmniHSS 的部署架构。

部署图



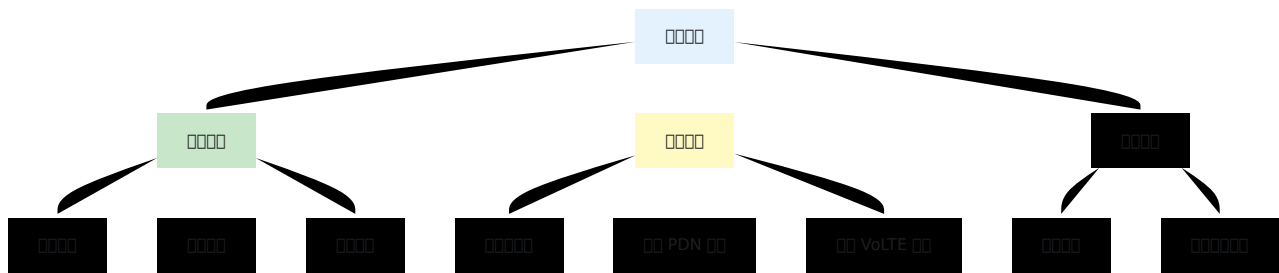
网络架构

网络架构概述

网络架构

URL: [https://\[hostname\]:7443/overview](https://[hostname]:7443/overview)

网络架构



网络架构

网络	网络	网络
网络	网络	网络
网络	MME 网络	网络
PDN 网络	PDN 网络 > 0	网络
IMS 网络	S-CSCF 网络	网络
网络	网络 > 0	VoLTE 网络

网络架构

网络架构

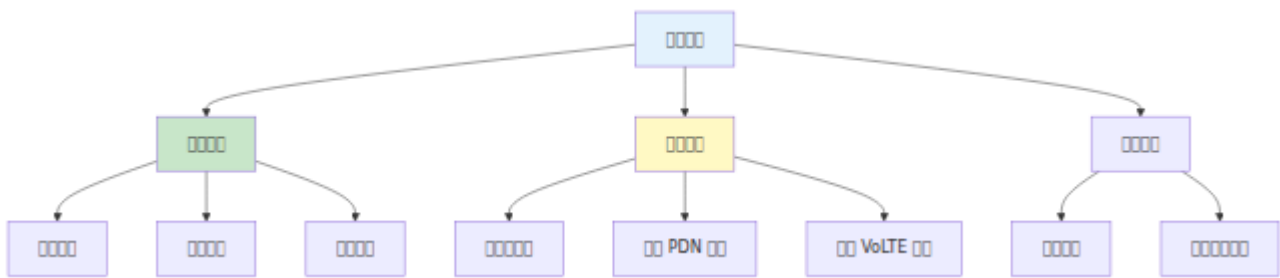
- 网络架构
- 网络架构
- 网络架构

4. Diameter 프로토콜

Diameter 프로토콜

URL: `https://[hostname]:7443/diameter`

개요



구성

구성 요소

구성 요소	기능	설명
MME	UE	UE와 MME 간의 LTE 연결
P-GW	UE	UE와 P-GW 간의 연결
S-CSCF	UE	UE와 S-CSCF 간의 IMS 연결
P-CSCF	UE	UE와 P-CSCF 간의 VoLTE 연결
I-CSCF	UE	IMS 연결
AS	UE-UE	UE와 UE 간의 연결

구성

URL: `https://[hostname]:7443/application`

개요

項目	単位	測定項目	測定結果
通話率	単位 Erlang 単位	通話率	> 90% 単位
通話率	通話率	< 80%	> 90%
通話率	通話率	単位	単位

通話率

通話率

単位 SQL 単位

単位

単位

- 単位
- 単位
- IMS 単位

単位

単位

- 単位 PDN 単位
- 単位 VoLTE 単位
- 単位 APN 単位 PDN 単位

単位

単位

- 単位 MCC-MNC 単位
- 単位 PLMN 001-001 単位

- 0000000000000000

0000

0000000000

- 0000000000000000
- 000 MME 00000000
- 0000000000000000

00000000

000000000000

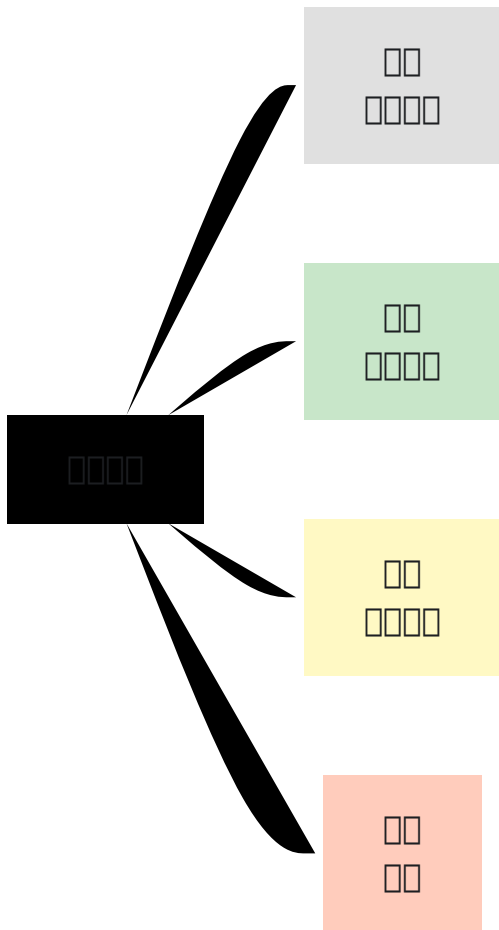
- 0000000000000000
- 0000000000
- 0000000000
- 000000000000

0000

0000

OmniHSS 000000 **stdout/stderr**00000000000000

0000



Network Configuration

Diameter Configuration:

```
[info] Diameter peer connected: mme01.epc.example.com
[warn] Diameter peer disconnected: pgw01.epc.example.com
[error] Diameter peer connection failed: timeout
```

Database:

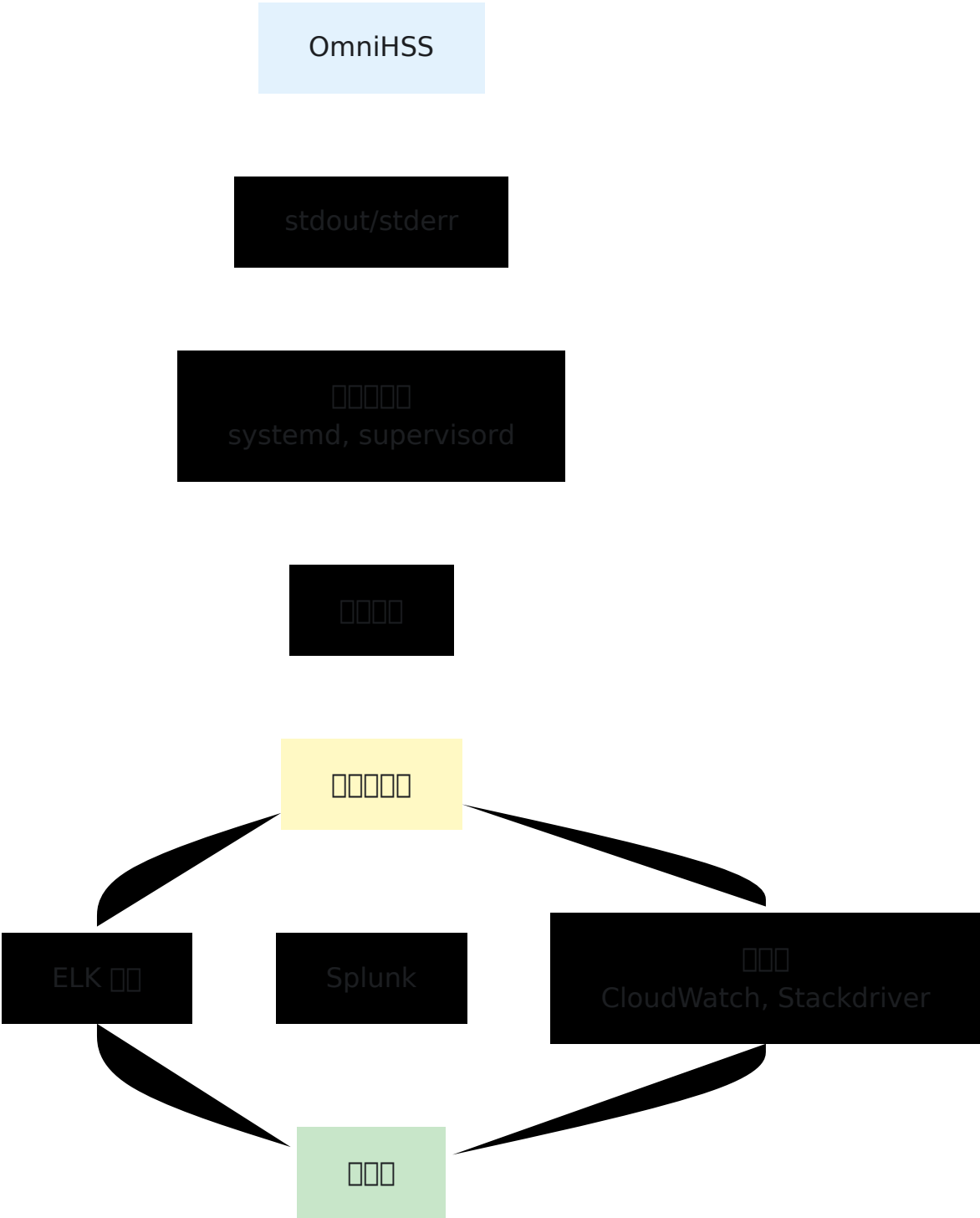
```
[info] Database connection established
[error] Database connection lost: timeout
[error] Database query failed: deadlock detected
```

Logs:

[info] Authentication successful: IMSI 001001123456789
[warn] Authentication failed: IMSI 001001123456789, invalid vector
[error] Roaming denied: IMSI 001001123456789, MCC 310 MNC 410

0000

0000000000000000



□□□□□□

□□□□□□

API □□□□: GET /api/status

```
curl -k https://hss.example.com:8443/api/status
```

□□□□:

```
{"status": "ok"}
```

HTTP □□: 200 OK

□□□□□□

Nagios/Icinga □□

```
#!/bin/bash
# check_omnihss.sh

API_URL="https://hss.example.com:8443/api/status"

response=$(curl -k -s -o /dev/null -w "%{http_code}" "$API_URL" --max-time 5)

if [ "$response" = "200" ]; then
    echo "OK - OmniHSS API responding"
    exit 0
else
    echo "CRITICAL - OmniHSS API not responding (HTTP $response)"
    exit 2
fi
```

Prometheus □□

OmniHSS Prometheus API

SNMP

SNMP SNMP API SNMP OID

KPI

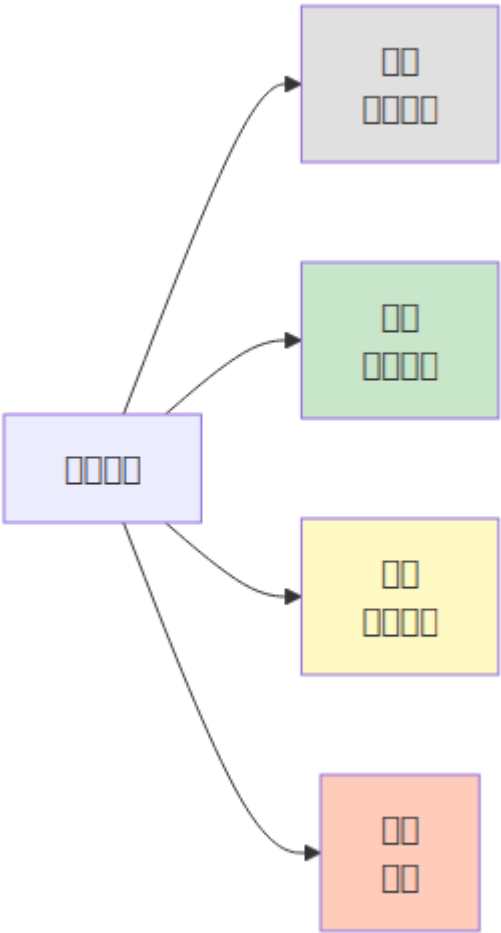


Table 1: KPI

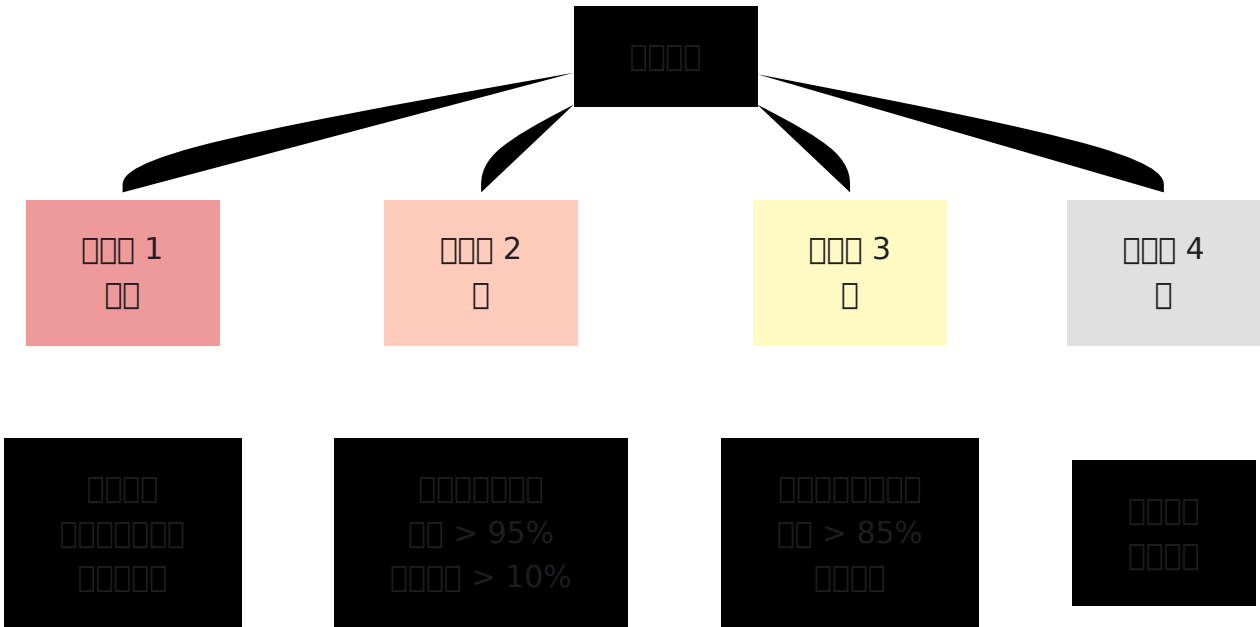
KPI			
	99.99%	< 99.95%	< 99.9%
Diameter	99.9%	< 99.5%	< 99%
	> 99%	< 99%	< 95%
Diameter	< 100ms	> 200ms	> 500ms
	< 50ms	> 100ms	> 500ms
	< 0.1%	> 0.5%	> 1%

Table 2: KPI

		80%
PDN		70%
	MB	80%
		80%

□□□□

□□□□□



□□□□

□□□□ (P1)

□□□□□:

- API □□□□□□
- □□□□□□□□
- □□□□□□□
- □□: □□□□□□□□

□□ **Diameter** □□□□□:

- □□□□□□
- □□: □□□□□□□□□□□□

□□□□□:

- □□□□□ SQL □□□

- 測試: 測試測試測試測試測試測試

測試結果 (P2)

測試 Diameter 測試結果:

- 測試 MME 測試
- 測試 P-GW 測試
- 測試 S-CSCF 測試
- 測試: 測試 15 測試測試測試測試

測試結果:

- 測試 > 95%
- 測試: 測試測試測試測試測試

測試結果:

- | 10% 測試測試測試
- 測試: 測試測試測試測試測試

測試結果 (P3)

測試測試測試:

- 測試測試測試
- 測試測試測試
- 測試: 測試 1 測試測試

測試結果:

- 測試 > 85%
- 測試: 測試測試測試測試測試

測試結果:

- 測試 > 1%
- 測試: 測試測試測試測試測試

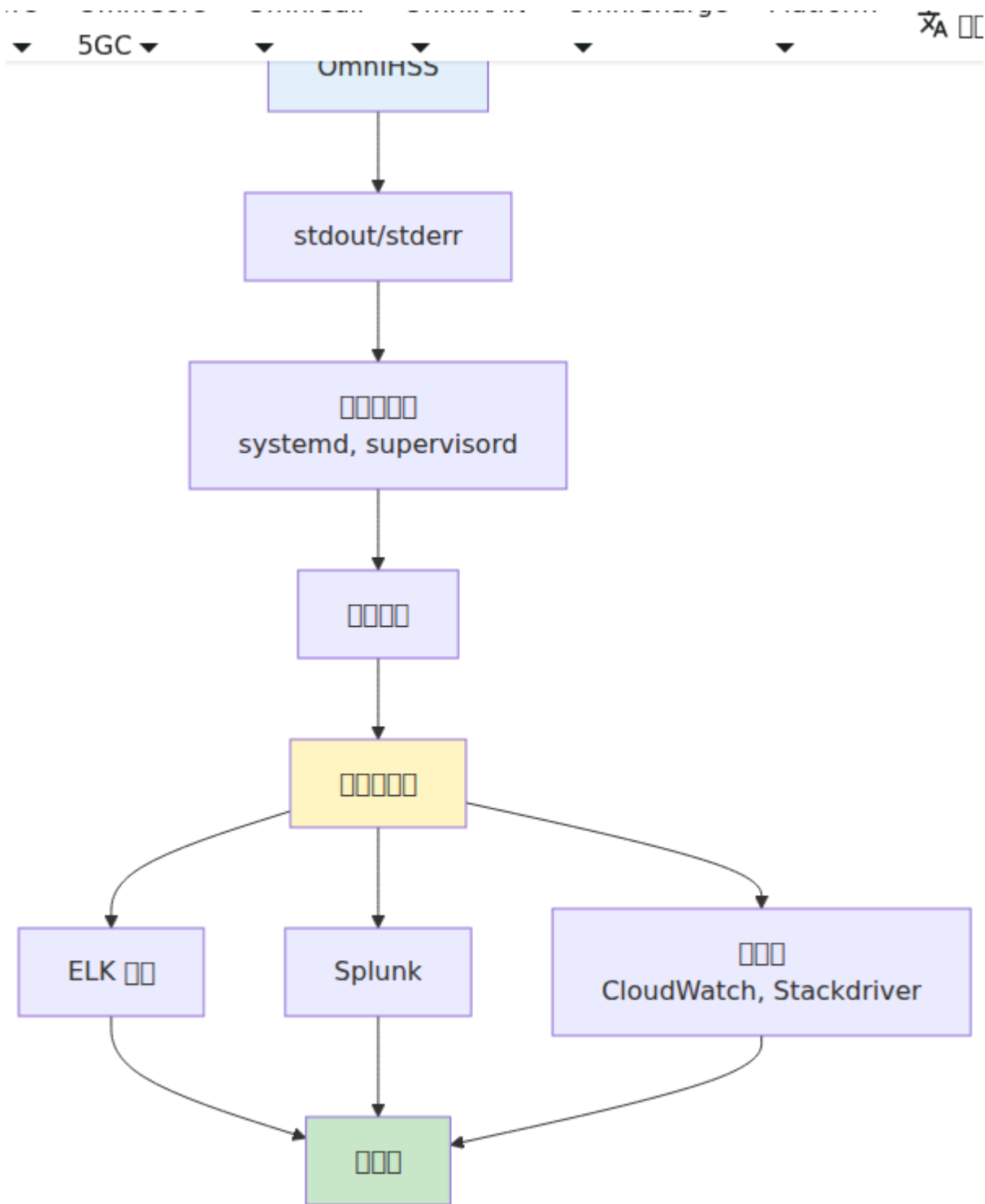
□□□□□ (P4)

□□□□:

- □□ > 80% □□□
- □□□ > 80% □□□□□
- □□: □□□□□□

□□□□:

- □□□□□□□□□□
- □□: □□□□□□□□



□□□□□□

□□□□

- ☐ □□□□□□□ - □□□□□□
- ☐ □□ Diameter □□ - □□□□□□□□□□
- ☐ □□□□□□ - □□□□□□□□□□
- ☐ □□□□□□ - □□ 24 □□□□□□□□
- ☐ □□□□□□□□

□□□□

- ☐ □□□□□□ - □□□□
- ☐ □□□□□□ - □□□□
- ☐ □□□□□□□ - □□□□□□
- ☐ □□□□□ - □□□□
- ☐ □□□□□□ - □□□□□□

□□□□

- ☐ □□□□□□ - □□ 6 □□□
- ☐ □□□□□□ - □□□□□□
- ☐ □□□□ - □□□□□□□□
- ☐ □□□□ - □□□□□□
- ☐ □□□□□□ - □□□□□□□□

OmniHSS MSISDN IMSI

←

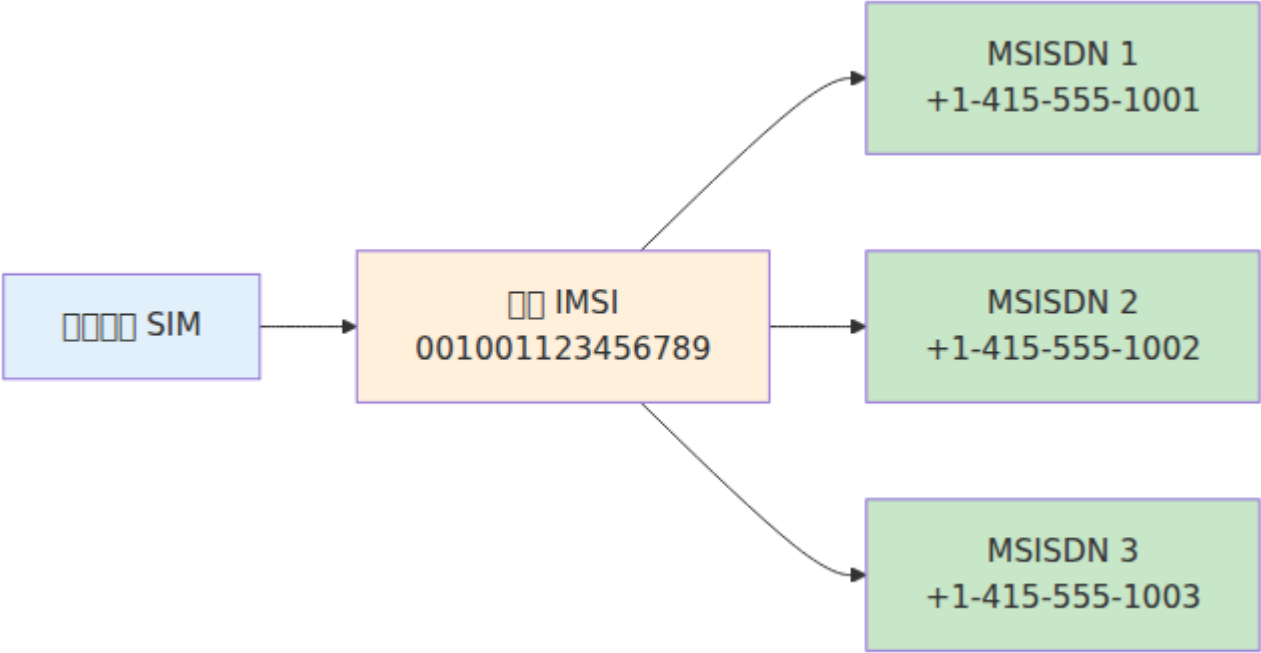
-
- MSISDN
- IMSI SIM
-
-
-

OmniHSS

MSISDN

IMSI →

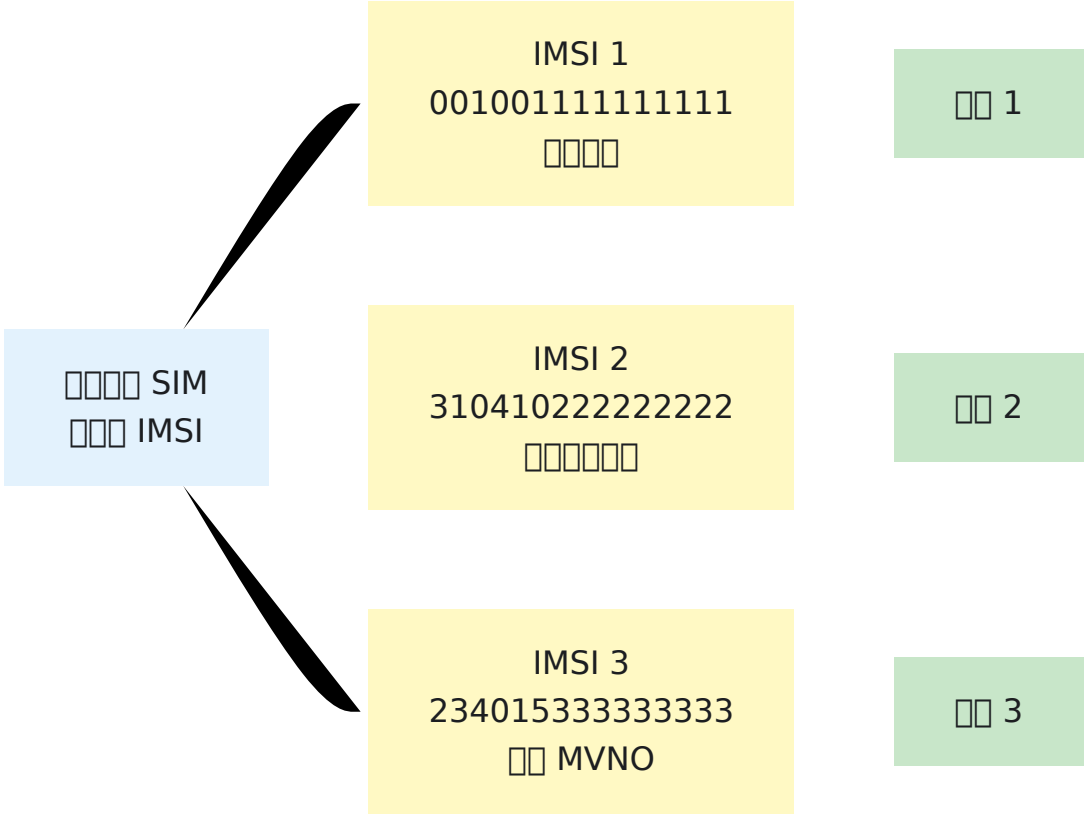
IMSI MSISDN



IMSI SIM

SIM → IMSI

SIM IMSI MVNO



MSISDN

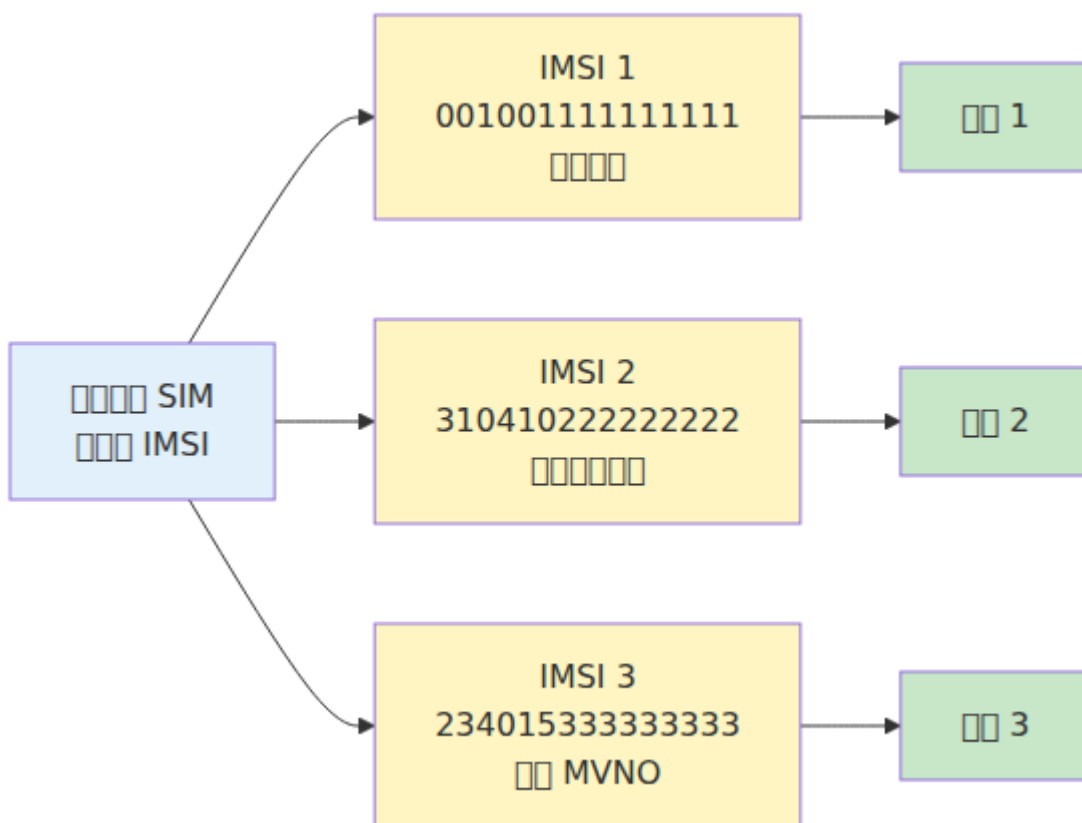
MSISDN

HSS MSISDN IMS MSISDN IMS

MSISDN

- IMSI - SIM IMSI
- MSISDN -
- IMS - MSISDN IMS
- EPC IMS

MSISDN

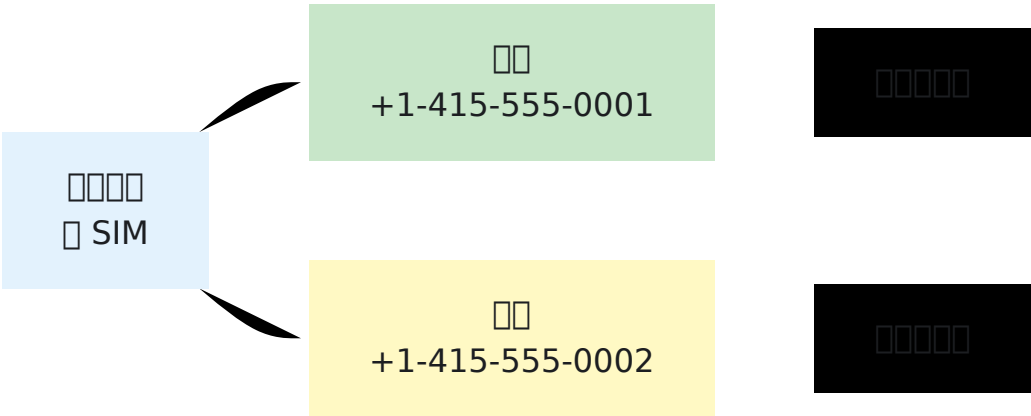


MSISDN MSISDN

□□□□

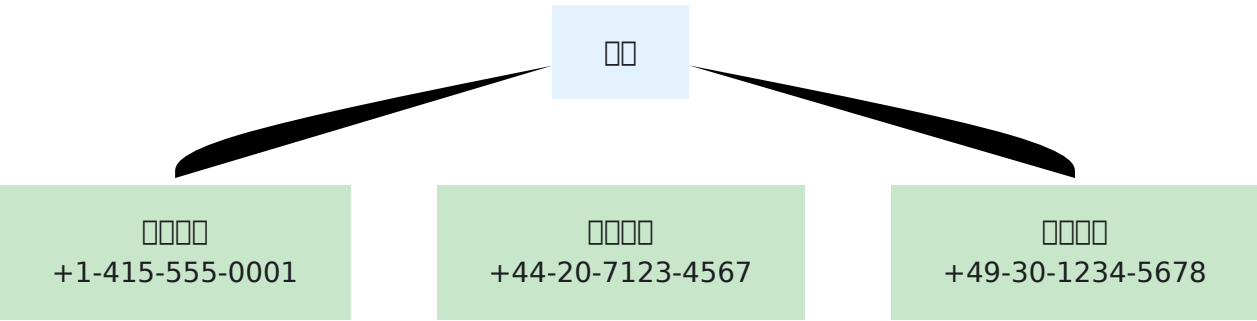
1. □□□□□□

□□□□□□□□□□□□□□□□□□



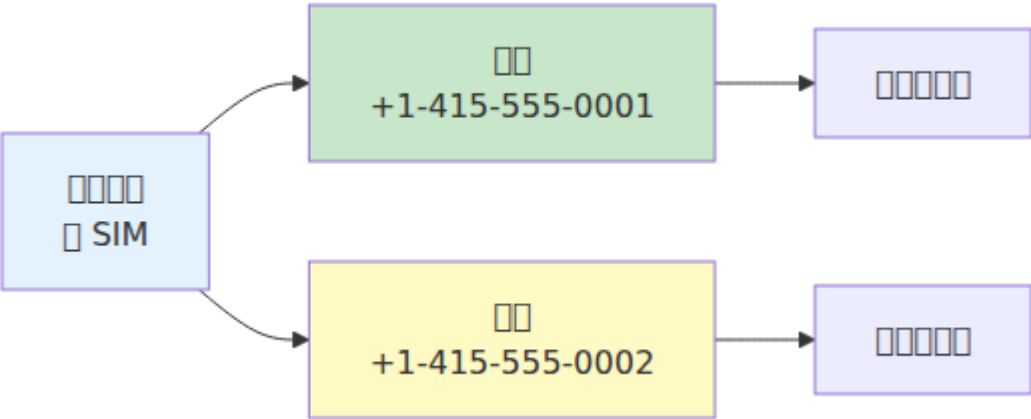
2. □□□□

□□□□□□□□□□□□□□□□



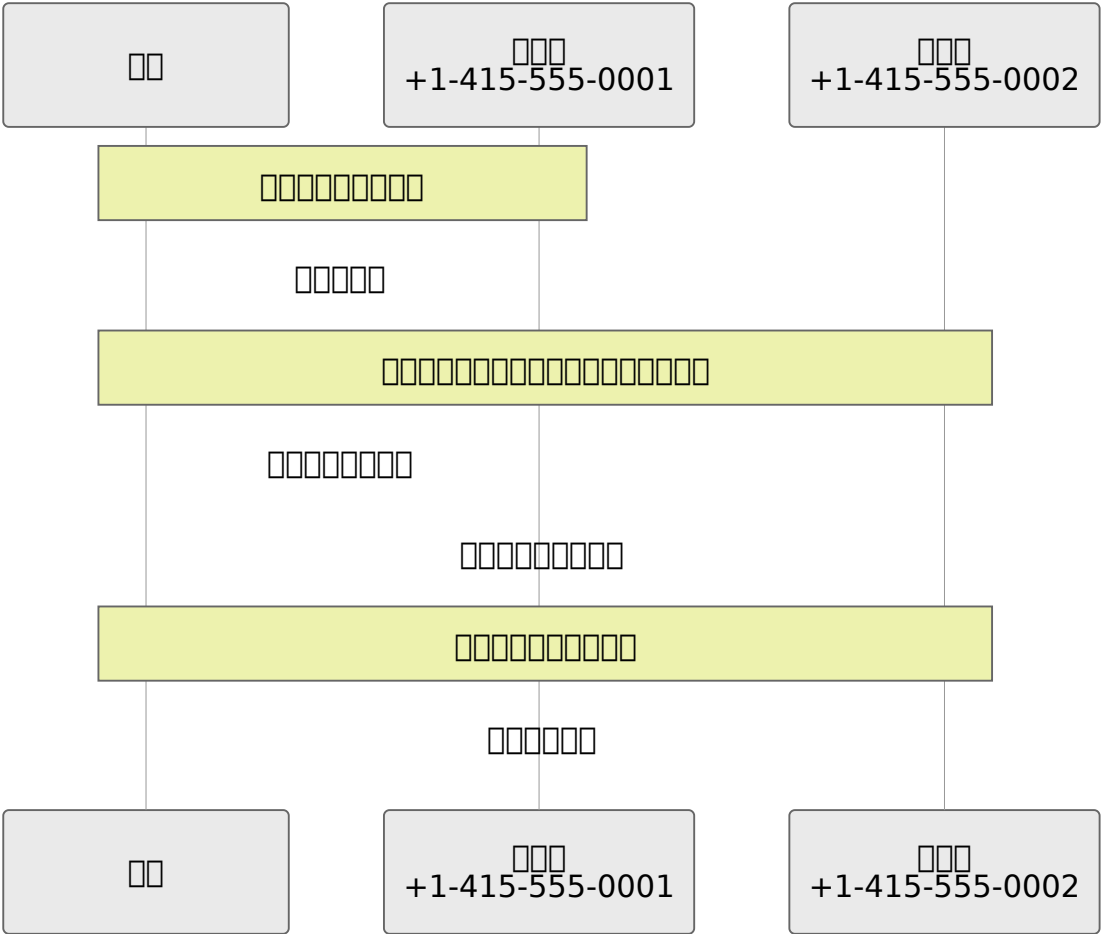
3. □□□□

□□□□□□□□□□□□□□



OmniHSS SIM/IMSI MSISDN

4.



MSISDN

MSISDN

```
# 创建 MSISDN
curl -k -X POST https://hss.example.com:8443/api/msisdn \
-H "Content-Type: application/json" \
-d '{"msisdn": {"msisdn": "14155551001"}}'

# 创建 MSISDN
curl -k -X POST https://hss.example.com:8443/api/msisdn \
-H "Content-Type: application/json" \
-d '{"msisdn": {"msisdn": "14155551002"}}'
```

MSISDN 管理

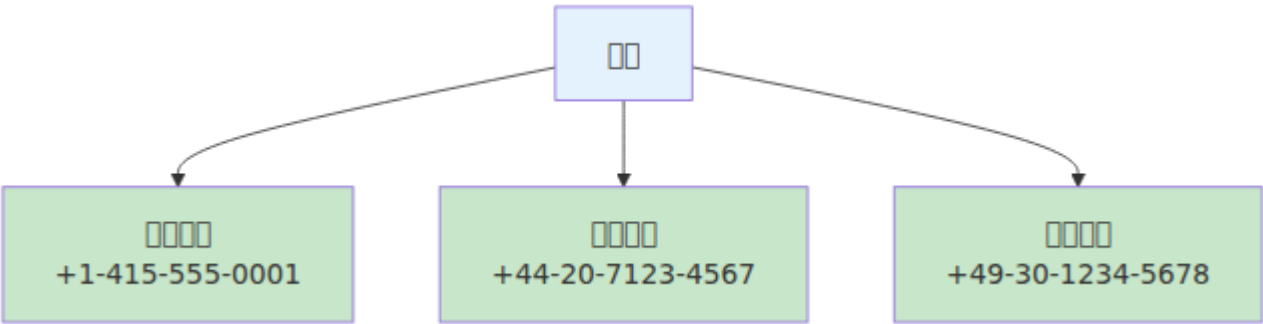
管理 MSISDN 列表

管理 MSISDN

- 1. 创建 MSISDN 时 IMSI 和 ID
- 2. 创建 MSISDN 时 MSISDN ID
- 3. 创建 MSISDN 时 subscriber_id 和 msisdn_id

管理 MSISDN 列表

管理 MSISDN



管理 MSISDN

管理 MSISDN 列表

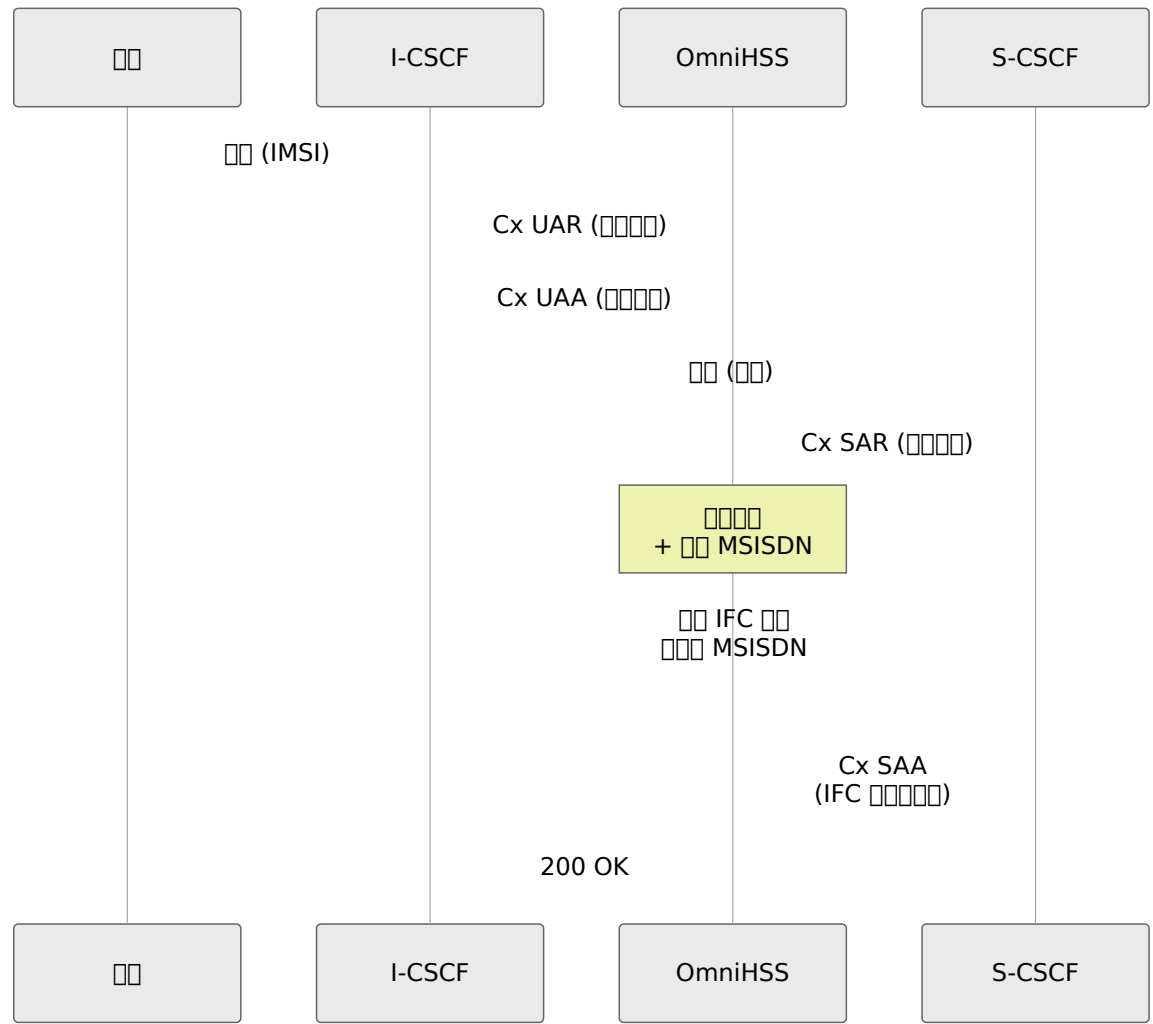
- 创建 MSISDN
- 创建 msisdn 列表
- 创建 MSISDN 列表

□□□□□ ID□IMSI □□□□□ MSISDN □□□

IMS □□

IMS □□

□□□□□ IMS □□□□□□ **MSISDN** □□□□□□ S-CSCF □ IMS □□□□□□



IFC □□□□

IMS IFC □□□□□ `{{msisdns}}` □□□□□ MSISDN□

□□ **IFC** □□□

```

<ServiceProfile>
  <PublicIdentity>
    <Identity>sip:
{{imsi}}@ims.mnc{{mnc}}.mcc{{mcc}}.3gppnetwork.org</Identity>
  </PublicIdentity>
  <!-- MSISDN -->
  <PublicIdentity>
    <Identity>sip:+14155551001@ims.example.com</Identity>
  </PublicIdentity>
  <PublicIdentity>
    <Identity>tel:+14155551001</Identity>
  </PublicIdentity>
  <PublicIdentity>
    <Identity>sip:+14155551002@ims.example.com</Identity>
  </PublicIdentity>
  <PublicIdentity>
    <Identity>tel:+14155551002</Identity>
  </PublicIdentity>
  <!-- ... -->
</ServiceProfile>

```

□□□□□

- `{{msisdns}}` - □□□□□□□□ MSISDN □□

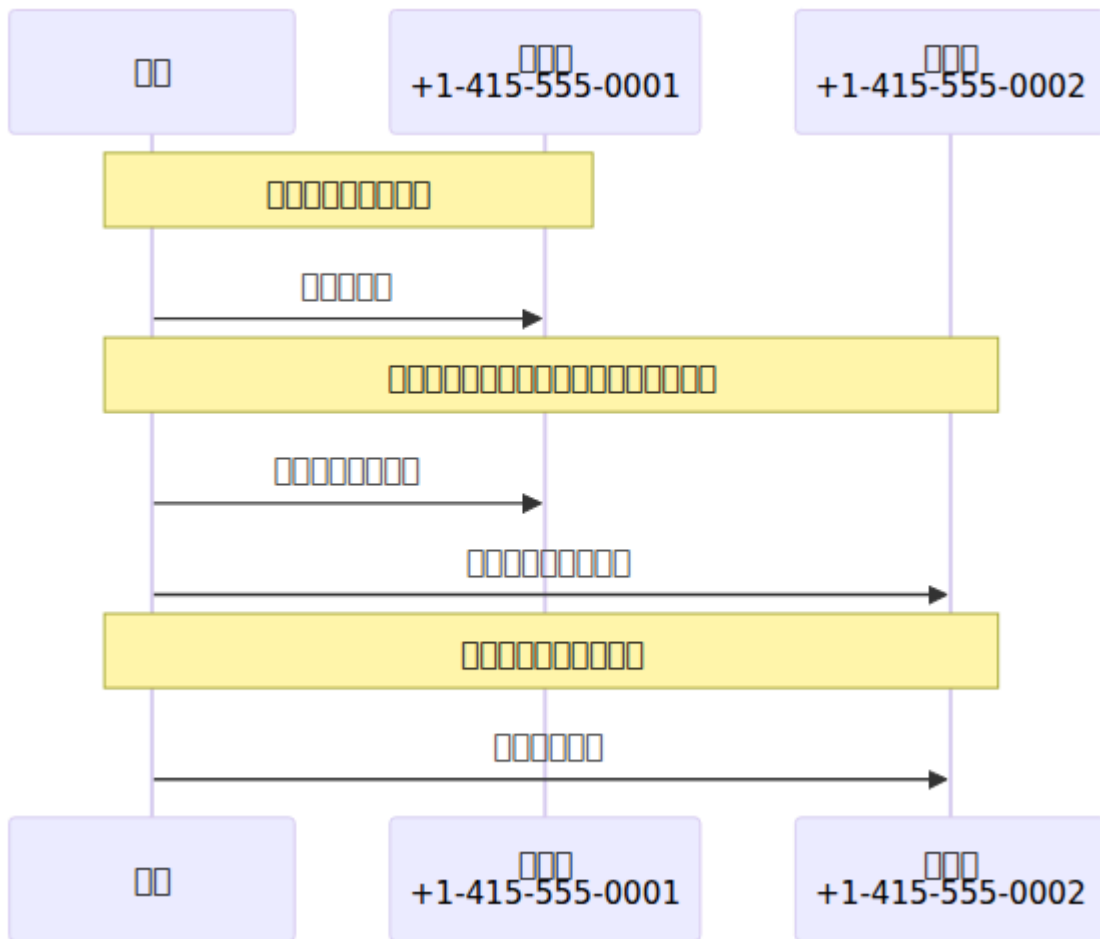
□□□□

□□ MSISDN □□□□□□□□ IMS □□□□□



□□□□□□

□□□□□□□□□□□□□□ IMS □□□□□□□□□□ SIP URI□



INVITE sip:+15105551234@ims.example.com SIP/2.0

From: "+14155551002" <sip:+14155551002@ims.example.com>;tag=123

To: <sip:+15105551234@ims.example.com>

P-Asserted-Identity: <sip:+14155551002@ims.example.com>

From: P-Asserted-Identity

MSISDN

MSISDN IMS

- S-CSCF 認證與授權
- 認證與授權

認證與授權

1. 認證與授權 **MSISDN** 認證

- 認證與授權時 IMSI 認證與 **MSISDN**
- 認證與授權時

2. 認證 **IMS** 認證與授權

- 認證與授權時 `{{msisdns}}` 認證
- 認證與授權時 XML

3. 認證 **HSS** 認證

- 認證 IMS 認證 Cx SAR 認證
- 認證 **MSISDN** 認證與授權

4. 認證 **IMS** 認證

- 認證與授權時
- 認證 S-CSCF 認證與授權

認證與授權 **MSISDN** 認證與授權

認證

- 認證與授權
- 認證“認證”與“認證”

認證與授權

1. **MSISDN** 認證與授權

-  認證與授權 **MSISDN** 認證與授權
- 認證與授權 認證與授權

2. **MSISDN** 認證與授權

- 註冊號碼 MSISDN 號碼
- 號碼 的 API 註冊號碼 MSISDN 的

註冊號碼

的

- 註冊號碼
- 註冊號碼

註冊號碼

1. 的 IMS 註冊號碼

- 的 S-CSCF 註冊號碼
- 的 SIP URI 的

2. 的 IMS 註冊

- 的 IFC 註冊號碼
- 註冊號碼

3. 註冊號碼

```
# 的 SIP 註冊號碼
sip:+14155551001@ims.example.com # 的
sip:+14155551002@ims.example.com # 的
```

的 API 的 MSISDN 註冊號碼

的

- API 的 `/api/subscriber/msisdn/:msisdn` 註冊號碼

的

註冊號碼 MSISDN 註冊號碼

□□□□

□□□□

1. □◆◆□□□□ MSISDN
2. □□□□
3. □ MSISDN □□□□□
4. □□□□□□□□□

MSISDN □□

- □□□□ **custom_attributes** □□□□□□□□□□
- □□□□□□□□□□□□□□□□□□
- □□□□□□□□□□□□□□□□

IMS □□

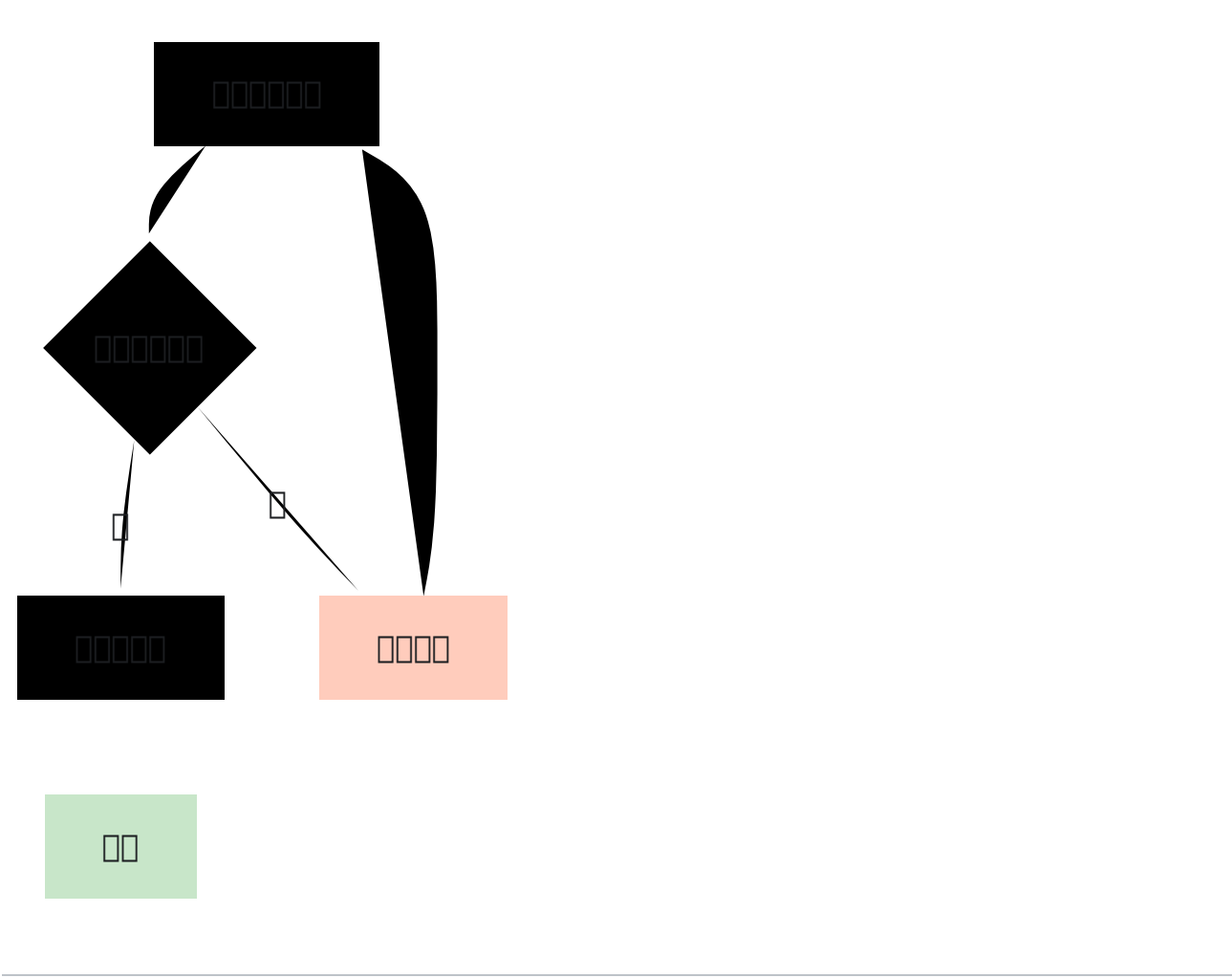
- □□ IFC □□□□□□□□□□□□□□
- □□□□□□□□□□□□□□□□
- □□□□□□□□□□□□□□□□

□□

□□□□□ MSISDN □□□□□

□□□ 1 □ MSISDN

MSISDN



IMSI SIM

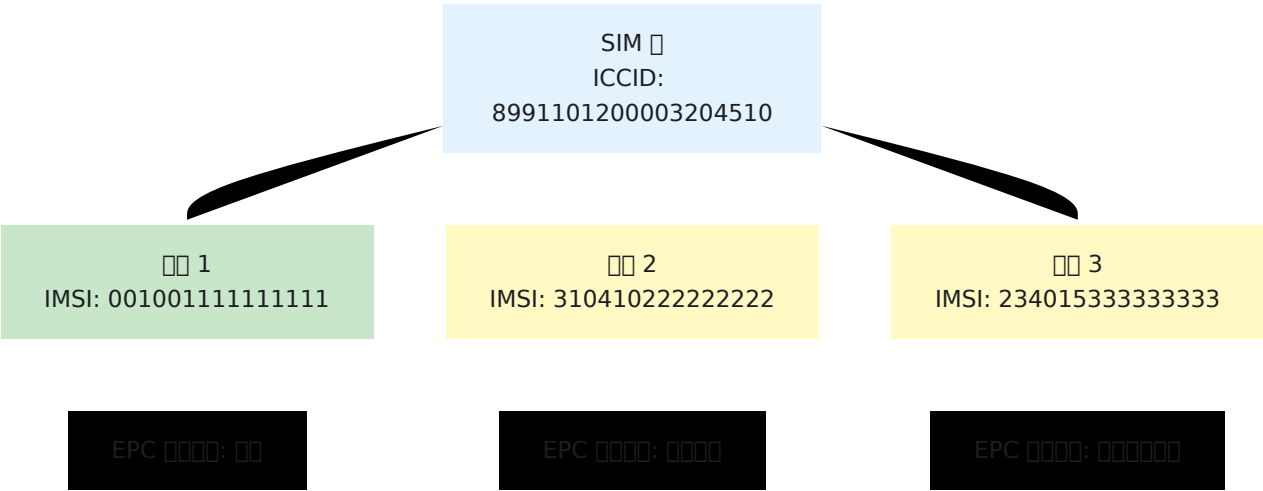
□□□□

IMSI SIM IMSI IMSI

物联网 物联网卡 IMSI 物联网卡 SIM 物联网卡 IMSI 物联网 HSS 物联网 物联网 物联网 物联网
 IMSI

OmniHSS 物联网

物联网 OmniHSS 物联网 IMSI SIM 物联网 IMSI 物联网 物联网 物联网 物联网 物联网 SIM 物联网



物联网

1. 物联网

- 物联网 IMSI: 001-001物联网
- 物联网 IMSI: 310-410物联网
- 物联网 IMSI: 234-015物联网
- 物联网 IMSI

2. MVNO 物联网

- 物联网 IMSI: MVNO 物联网
- 物联网 IMSI: 物联网
- 物联网 MVNO 物联网

3. IoT/M2M 物联网

- IMSI 1: 物联网
- IMSI 2: 物联网

- □□□□□□□

4. □□□□

- □□□□□□◆◆◆□□□□□

IMS I

□□□□

- [illegible]

□□□□□□□□

- IMSI

□ □ □ □ □ □

- SIM 00000000

□□□□

- HSS □□□□□□□□ IMSI



```
# 1. SIM 1 IMSI
SIM_ID=$(curl -k -X POST https://hss.example.com:8443/api/sim \
  -d '{"sim": {"iccid": "8991101200003204510", "is_esim": false}}' \
  | jq -r '.data.id')
```

```
# 2. IMSI 1 KEYSET
KEYSET1=$(curl -k -X POST https://hss.example.com:8443/api/key_set \
  -d '{"key_set": {"ki": "0123456789ABCDEF...", "opc": "FEDCBA9876..."}}' \
  | jq -r '.data.id')
```

```
# 3. SIM 1 IMSI
curl -k -X POST https://hss.example.com:8443/api/subscriber \
  -d '{"subscriber": {
    "imsi": "\"001001111111111\"",
    "sim_id": "\"$SIM_ID\"",
    "key_set_id": "\"$KEYSET1\"",
    "epc_profile_id": "\"e58ed763-928c-4155-bee9-fdbaaadc15f3\""
  }}'
```

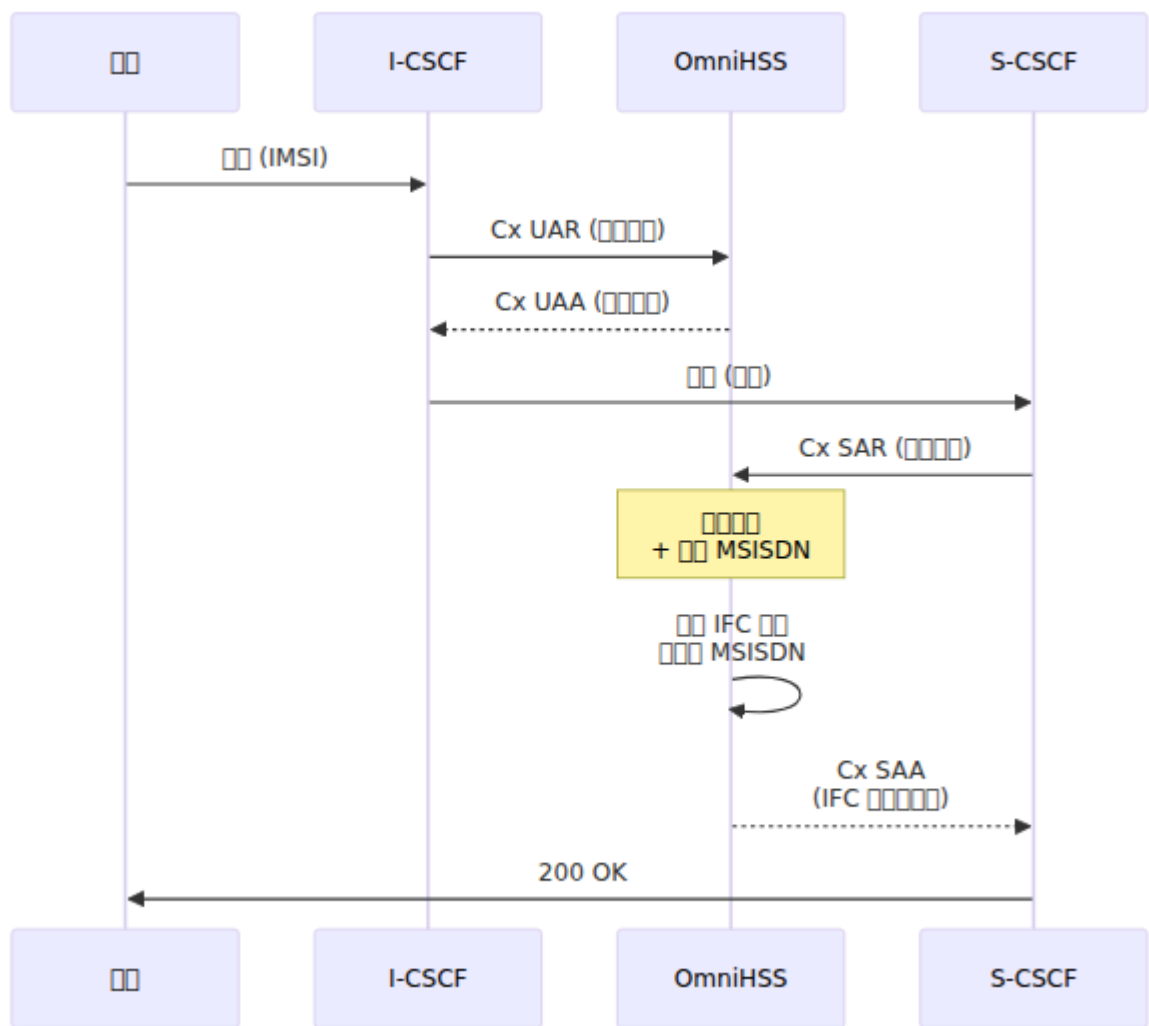
```
# 4. IMSI 2 KEYSET
KEYSET2=$(curl -k -X POST https://hss.example.com:8443/api/key_set \
  -d '{"key_set": {"ki": "1111111111111111...", "opc": "2222222222..."}}' \
  | jq -r '.data.id')
```

```
# 5. SIM 2 IMSI
curl -k -X POST https://hss.example.com:8443/api/subscriber \
  -d '{"subscriber": {
    "imsi": "\"310410222222222\"",
    "sim_id": "\"$SIM_ID\"",
    "key_set_id": "\"$KEYSET2\"",
    "epc_profile_id": "\"f47ac10b-58cc-4372-a567-0e02b2c3d479\""
  }}'
```

```
# 6. SIM 2 IMSI 222222...
```

□□□□

□□ IMSI □□□□□□



HSS □□□□□□□□□□ IMSI SIM□□□□□□□□□□ IMSI□

IMSI □□□□□□□□

□□ IMSI SIM □□□ IMSI □□□□□□ IMSI □□□□□□□□□□ IMSI□OmniHSS □□□ HSS □□**□□□□
□□□□CLR□**□□□□□□□□□□□□□□□□□□□□□□ IMSI□□□□ SIM □□□ IMSI □□□□

□□□□ **IMSI** □□

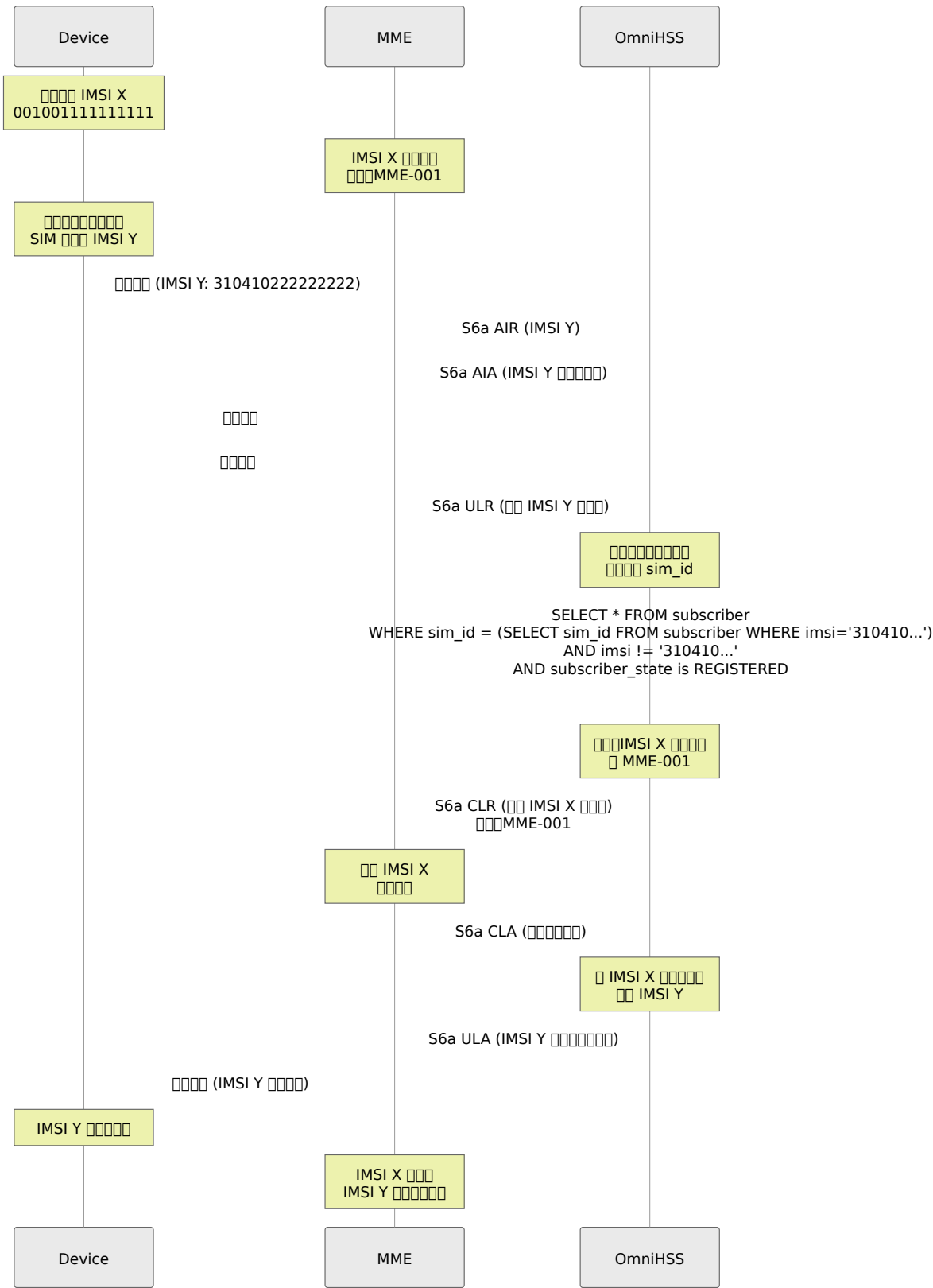
□□□□□ □□ SIM □□□□□□□□□□IMSI□□□□□□□□□□

- □□□□□ MME □□□**IMSI X**□□
- □□ HSS □□□□**IMSI Y**□□□□□□□□□□ IMSI X □□□ SIM □□

- HSS 0000000000000000IMSI X

0000 IMSI 00000000000000000000

IMSI 0000



□□□□□□

□□□□□□

- □□□□□□□SIM□□□□□
- □□□□□□□□□□
- □□□□□□□□□□

□□□□□□

- □□□□□□IMSI□□□□□□□□
- IMSI□□□□□□□□□□
- □□□□□□□□CDR□□□

□□□□□□

- □IMSI□MME□□□□□
- PDP□□□□□□□□□□
- □□□□□□□□

IMSI□□□□□

□□/SIM□□□□□□IMSI□□□□

1. □□□□□

- □□IMSI□□□□□
- □□□□□□□□IMSI

2. □□□□

- □□□□□□□□
- SIM□□□□□□IMSI

3. □□□□

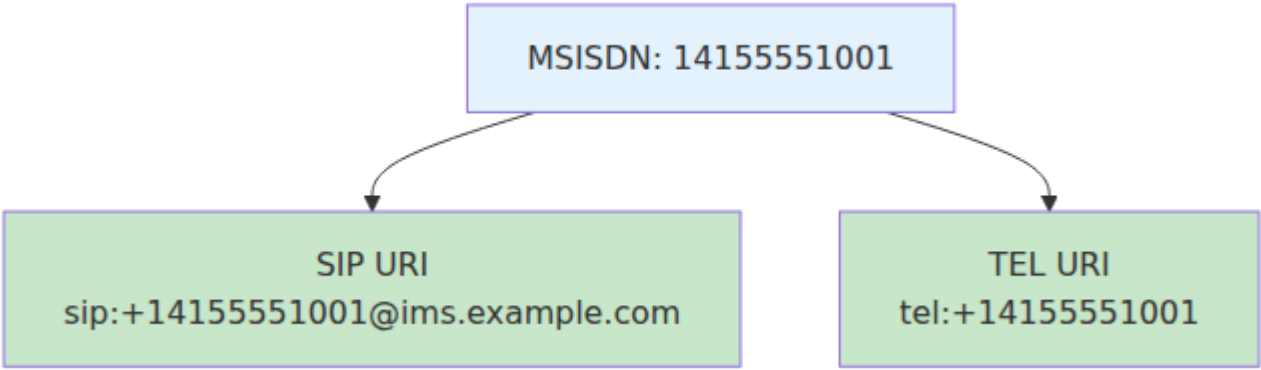
- SIM□□□□□□□□□□□□□□IMSI□
- □□MCC/MNC□□□□□

4. □□□□

- 國際移動網際網路識別碼IMSI
- 國際移動網際網路識別碼IMSI

IMS

國際移動網際網路識別碼IMSI



國際移動網際網路識別碼IMSI

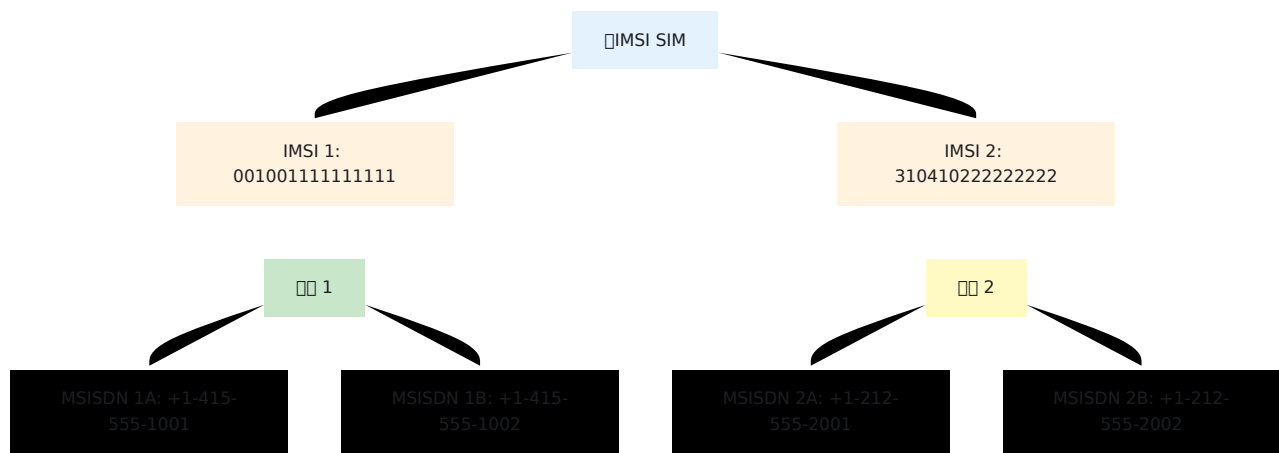
國際移動網際網路識別碼IMSI

1. 國際移動網際網路識別碼IMSI國際移動網際網路識別碼IMSIHSS“國際移動網際網路識別碼IMSI”
2. 國際移動網際網路識別碼IMSI SIM國際移動網際網路識別碼sim_id國際移動網際網路識別碼“國際移動網際網路識別碼IMSI”
3. 國際移動網際網路識別碼subscriber_state國際移動網際網路識別碼IMSI國際移動網際網路識別碼MME/SGSN國際移動網際網路識別碼IMSI國際移動網際網路識別碼IMSI
4. 國際移動網際網路識別碼國際移動網際網路識別碼
 - 國際移動網際網路識別碼IMSI
 - 國際移動網際網路識別碼IMSI國際移動網際網路識別碼IMSI
 - 國際移動網際網路識別碼SIM國際移動網際網路識別碼IMSI國際移動網際網路識別碼IMSI

國際移動網際網路識別碼IMSI

IMSI + MSISDN

國際移動網際網路識別碼SIM國際移動網際網路識別碼IMSI國際移動網際網路識別碼IMSI國際移動網際網路識別碼MSISDN



□□□□

- □□□□ **IMSI 1**□
 - □□□□+1-415-555-1001
 - □□□□+1-415-555-1002
- □□□□□□ **IMSI 2**□
 - □□□□+1-212-555-2001
 - □□□□+1-212-555-2002

□□□□□□□□□□IMSI 1□□MSISDN□□□□□□□□□□IMSI 2□□□□MSISDN□□□□□□□□□□

□□□□

□□□**MSISDN**□□

□□□□□□□MSISDN□

□□API□□□GET /api/subscriber/imsi/:imsi

□□□□□□□□□□MSISDN□

如何获取IMSI

如何获取IMSI

1. 通过IMSI获取手机号码
2. 通过key_set获取IMSI
3. 通过EPC获取IMSI
4. 通过其他途径获取

如何获取IMSI

- 通过SIM卡获取HSS
- HSS获取IMSI
- 通过IMSI获取

如何获取MSISDN

如何获取MSISDN

1. 通过MSISDN获取手机号码
2. 通过IMS获取{{msisdns}}
3. 通过IMS获取
4. 通过S-CSCF获取

如何获取MSISDN

- 通过SIM卡获取HSS
 - HSS获取IMSI
 - 通过IMSI获取
-

□□□□

□MSISDN□□

- ✓ □□SIM□□□□□□□□
- ✓ □□□□□□□□□□
- ✓ □□□□□□
- ✓ □□□□□□
- ✓ □□□□□□□□□□□□□□
- ✓ □□IMSI□□□□

□IMSI SIM□□

- ✓ □□□□□□
- ✓ □□□□□□
- ✓ □□□□□□□□
- ✓ □□□□□□
- ✓ □□□□□□
- ✓ □□□□□□□□

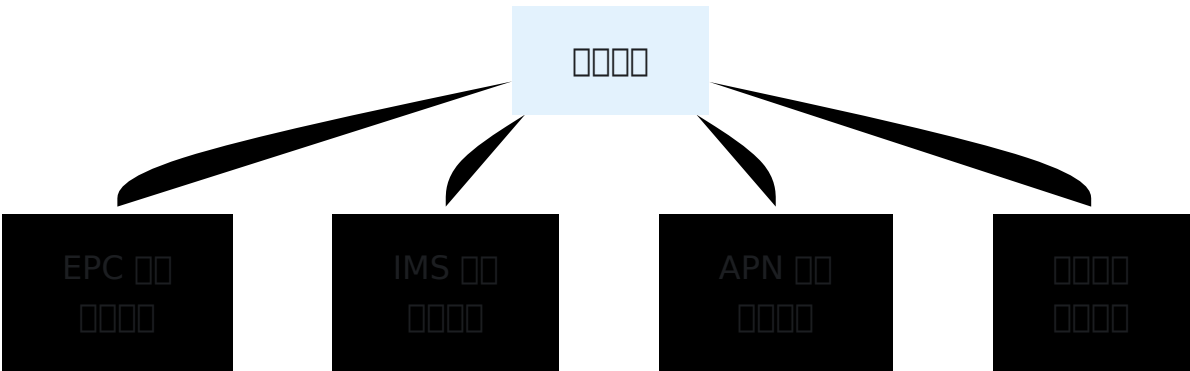
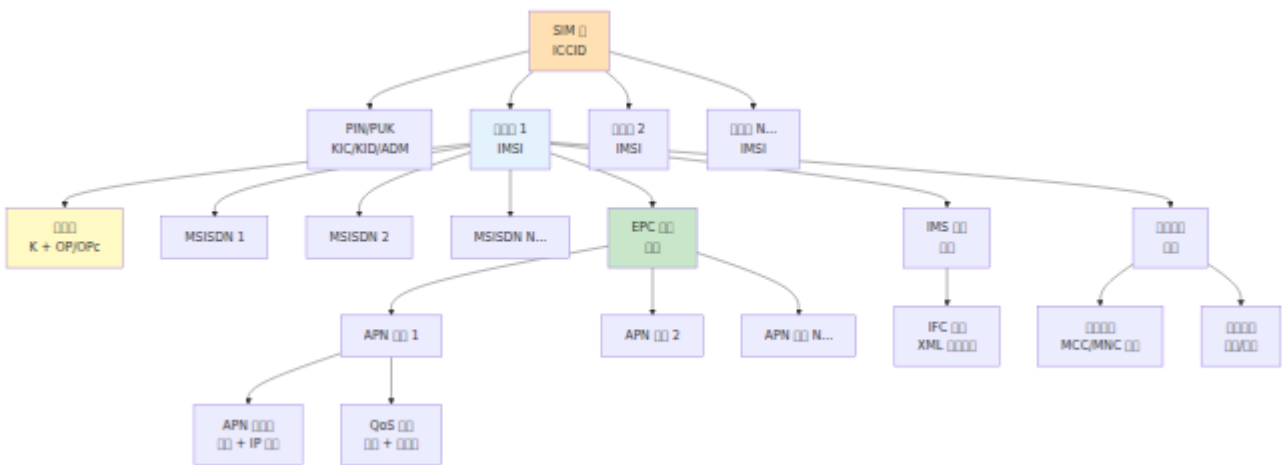
□□□□

- ✓ □□□□□□
- ✓ □□□□□□□□□□□□
- ✓ □□□□□□□□
- ✓ □□□□□□□□
- ✓ □□□□□□□□

OmniHSS

←

OmniHSS 是 一个 集中 式 的 网络 鉴权 和 计费 系统 ， 它 为 移动 网络 提供 安全 的 鉴权 和 计费 服务 。



EPC

EPCLTE

ue_ambr_dl_kbps	10,000 - 1,000,000 Kbps	
ue_ambr_ul_kbps	5,000 - 500,000 Kbps	
network_access_mode	"packet_only" or "packet_and_circuit"	
tracking_area_update_interval_seconds	TAU	54

EPC

```
curl -k -X POST https://hss.example.com:8443/api/epc/profile \
-H "Content-Type: application/json" \
-d '{
  "apn_profiles": [],
  "name": "Premium 100Mbps",
  "network_access_mode": "packet_only",
  "tracking_area_update_interval_seconds": 600,
  "ue_ambr_dl_kbps": 100000,
  "ue_ambr_ul_kbps": 50000
}'
```

EPC

- **CS**: 10 Mbps (10,000 Kbps)
- **PS**: 5 Mbps (5,000 Kbps)

CS:

- **CS**: 50 Mbps (50,000 Kbps)
- **PS**: 25 Mbps (25,000 Kbps)

PS:

- **CS**: 100 Mbps (100,000 Kbps)
- **PS**: 50 Mbps (50,000 Kbps)

PS:

- **CS**: 1 Gbps (1,000,000 Kbps)
- **PS**: 500 Mbps (500,000 Kbps)

IMS

IMS **IP** **Core** **Network** **Architecture** **IFC** **Interface** **Control** **Plane**

IFC

IFC **Interface** **Control** **Plane** **S-CSCF** **XML** **API**

IFC:

- **{{imsi}}** - IMSI
- **{{msisdns}}** - MSISDN
- **{{mcc}}** - MCC
- **{{mnc}}** - MNC

IMS

```
curl -k -X POST https://hss.example.com:8443/api/ims/profile \
-H "Content-Type: application/json" \
-d '{
  "ims_profile": {
    "name": "Standard VoLTE",
    "ifc_template": "<InitialFilterCriteria>...
</InitialFilterCriteria>"
  }
}'
```

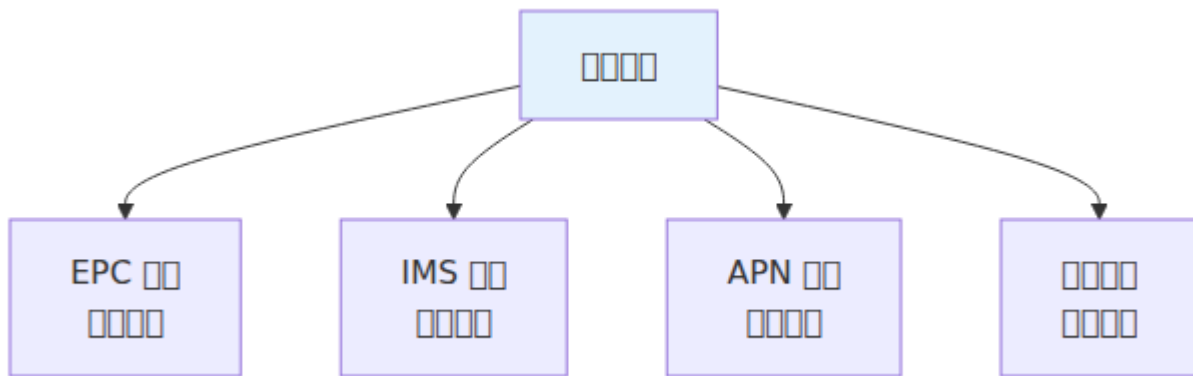
IFC

```
<ServiceProfile>
  <PublicIdentity>
    <Identity>sip:
{{imsi}}@ims.mnc{{mnc}}.mcc{{mcc}}.3gppnetwork.org</Identity>
  </PublicIdentity>
  <InitialFilterCriteria>
    <Priority>0</Priority>
    <TriggerPoint>
      <ConditionTypeCNF>0</ConditionTypeCNF>
      <SPT>
        <ConditionNegated>0</ConditionNegated>
        <Group>0</Group>
        <Method>INVITE</Method>
      </SPT>
    </TriggerPoint>
    <ApplicationServer>
      <ServerName>sip:as.ims.example.com</ServerName>
      <DefaultHandling>0</DefaultHandling>
    </ApplicationServer>
  </InitialFilterCriteria>
</ServiceProfile>
```

APN

APN

APN



APN

APN IP

APN:

- internet -
- ims - IMS/VoLTE
- mms -
- vzwadmin -

IP:

- "ipv4": IPv4
- "ipv6": IPv6
- "ipv4v6": IPv4v6
- "ipv4_or_ipv6": IPv4 or IPv6

APN QoS

QCI

QCI			
1	GBR		
2	GBR		
4	GBR		
5	GBR	IMS	
9	GBR		

APN QoS IP-CAN Gx CCA-I Online/Offline AVP

online_charging_enabled	boolean	null	Online AVP true = false = null PCRF PGW CCR-I
offline_charging_enabled	boolean	null	Offline AVP true = false = null PCRF PGW CCR-I

AVP PCRF:

📡📡📡📡 **APN** 📡📡

```
# 1. 📡📡 APN 📡📡
APN_ID=$(curl -k -X POST
https://hss.example.com:8443/api/apn/identifier \
-H "Content-Type: application/json" \
-d '{"apn": "internet", "ip_version": "ipv4v6"}' \
| jq -r '.response.id')

# 2. 📡📡 APN QoS 📡📡
QOS_ID=$(curl -k -X POST
https://hss.example.com:8443/api/apn/qos_profile \
-H "Content-Type: application/json" \
-d '{
  "name": "Best Effort",
  "allocation_retention_priority": 8,
  "apn_ambr_dl_kbps": 50000,
  "apn_ambr_ul_kbps": 25000,
  "pre_emption_capability": false,
  "pre_emption_vulnerability": true,
  "qci": 9
}' | jq -r '.response.id')

# 3. 📡📡 APN 📡📡
curl -k -X POST https://hss.example.com:8443/api/apn/profile \
-H "Content-Type: application/json" \
-d "{
  \"apn_identifier_id\": \"\$APN_ID\",
  \"apn_qos_profile_id\": \"\$QOS_ID\",
  \"name\": \"Internet APN\"
}"
```

📡 **APN** 📡📡📡 **EPC** 📡📡

APN 📡📡 `join_epc_profile_to_apn_profile` 📡📡 EPC 📡📡📡📡

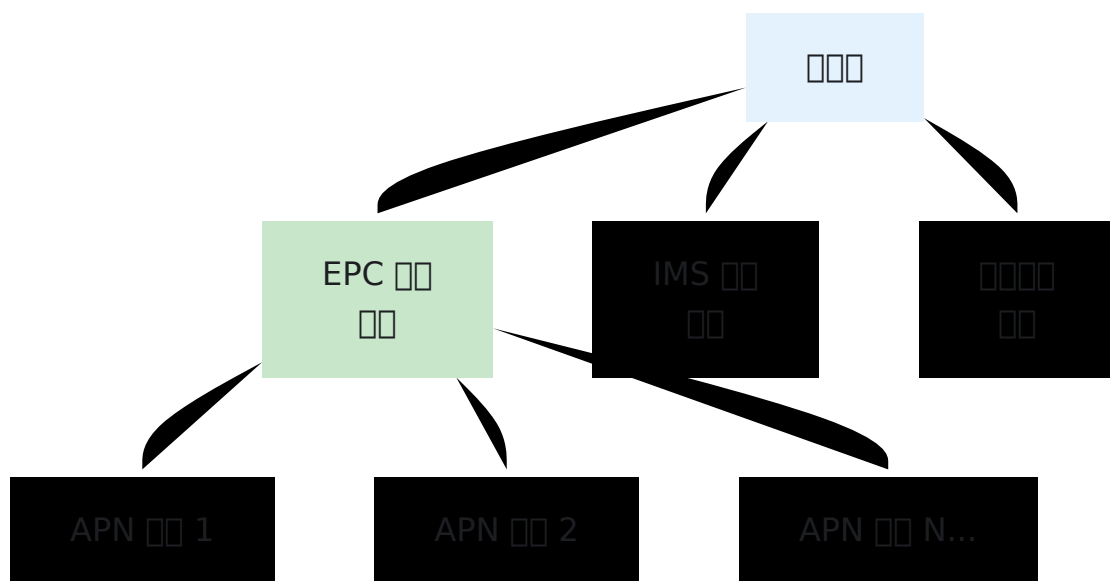
📡📡📡📡📡📡📡📡 APN 📡📡 ID 📡📡📡 EPC 📡📡 ID📡📡📡 APN 📡📡📡📡📡📡📡📡 EPC 📡📡📡

□□□□

□□□ □□□□□ □□□□□□

□□□□

□□□□□□□



API 測試

```
# 新增訂閱者 EPC 及 IMS 資料
curl -k -X POST https://hss.example.com:8443/api/subscriber \
-H "Content-Type: application/json" \
-d '{
  "subscriber": {
    "imsi": "001001123456789",
    "key_set_id": "6ba7b810-9dad-41d1-80b4-00c04fd430c8",
    "epc_profile_id": "e58ed763-928c-4155-bee9-fdbaaadc15f3",
    "ims_profile_id": "7c9e6679-7425-40de-944b-e07fc1f90ae7",
    "roaming_profile_id": "9c858901-8a57-4791-81fe-4c455b099bc9"
  }
}'

# 更新訂閱者 EPC 資料
curl -k -X PUT
https://hss.example.com:8443/api/subscriber/d290f1ee-6c54-4b01-
90e6-d701748f0851 \
-H "Content-Type: application/json" \
-d '{
  "subscriber": {
    "epc_profile_id": "f47ac10b-58cc-4372-a567-0e02b2c3d479"
  }
}'
```

測試結果

測試步驟

1. 新增訂閱者 - 新增訂閱者資料到 HSS
2. 更新訂閱者 - 更新訂閱者 EPC 資料
3. 刪除訂閱者 - 刪除訂閱者資料
4. 查詢訂閱者 - 查詢訂閱者資料

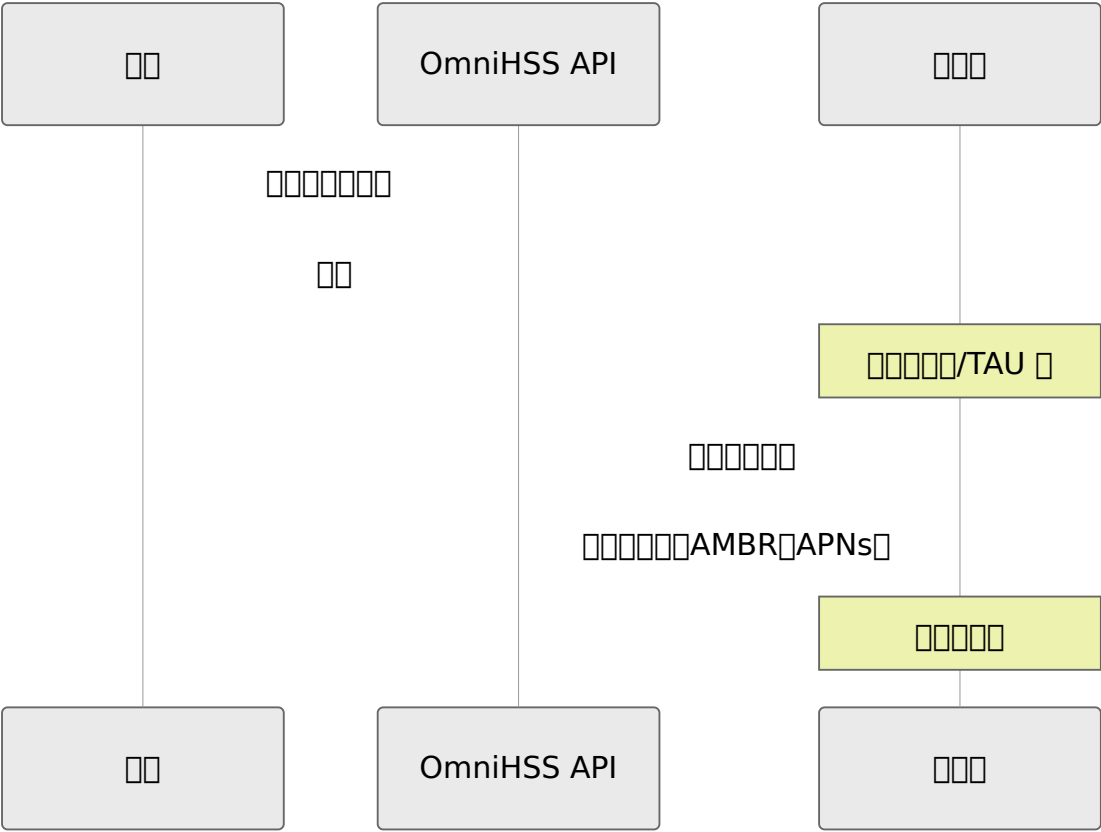
□□□□□□

[□□□□] - [□□] - [□□]

- :
- "Basic-10Mbps-Internet"
 - "Premium-100Mbps-VoLTE"
 - "Enterprise-1Gbps-MultiAPN"

□□□□

□□□□□□□□□□



□□: □□□□□□□□□□

- □□□□□□□TAU□
- □□
- IMS □□□□□ IMS □□□□□

配置参数

网络参数:

- 选择 EPC 的 AMBR
- 选择 APN QoS 的 AMBR
- 选择 MME/P-GW 的 QoS
- 选择

IMS 参数:

- 选择 IMS
- 选择 IFC 的 XML 文件
- 选择 S-CSCF 的 IFC 文件
- 选择 S-CSCF

APN 参数:

- 选择 APN 的 EPC
- 选择 APN 的 QoS
- 选择 UE 的 PDN

OmniHSS

← ☐ ☐ ☐ ☐ ☐ ☐

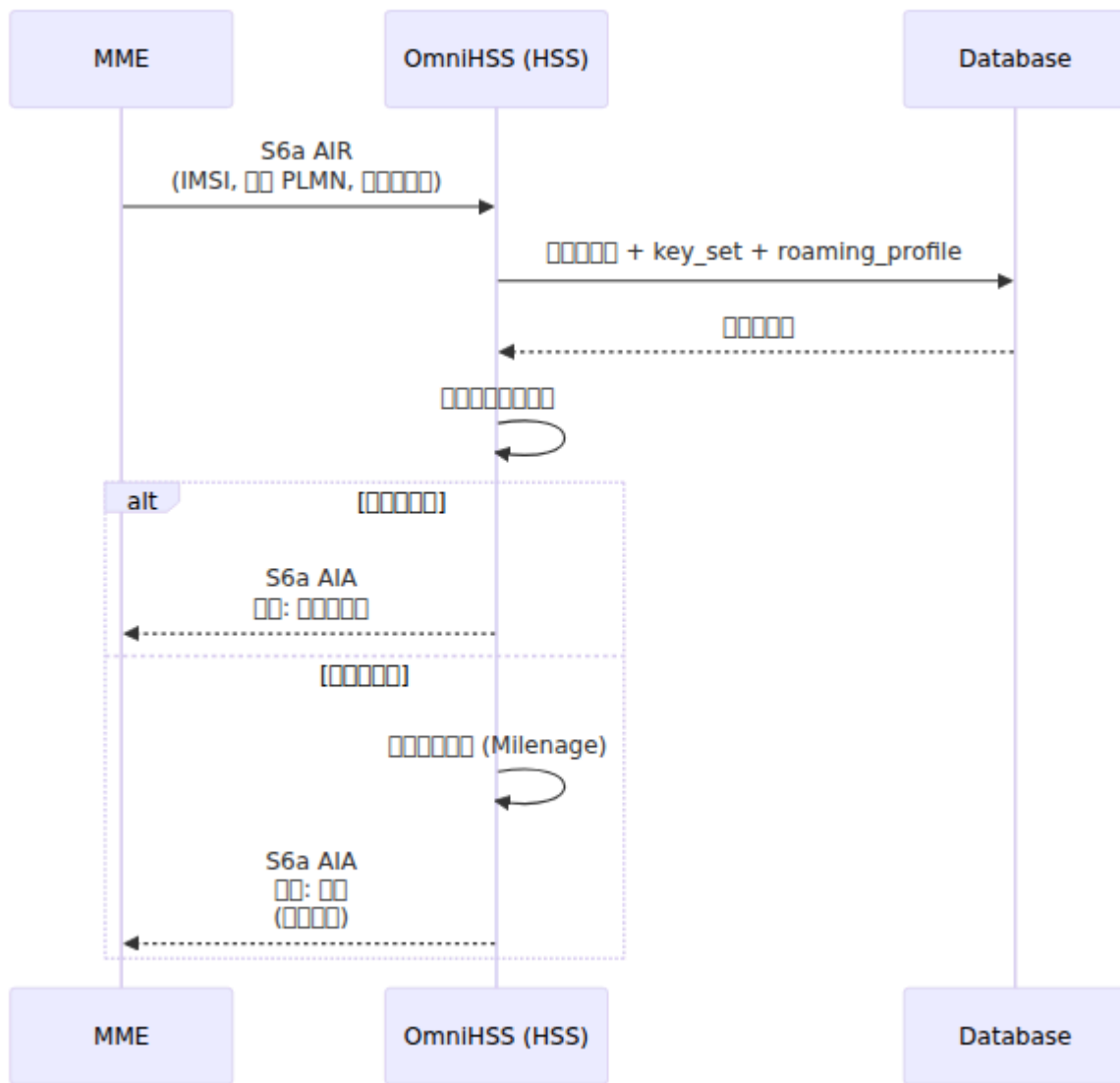
11

OmniHSS Diameter

S6a (LTE/EPC)

□□□□□ (AIR/AIA)

MME

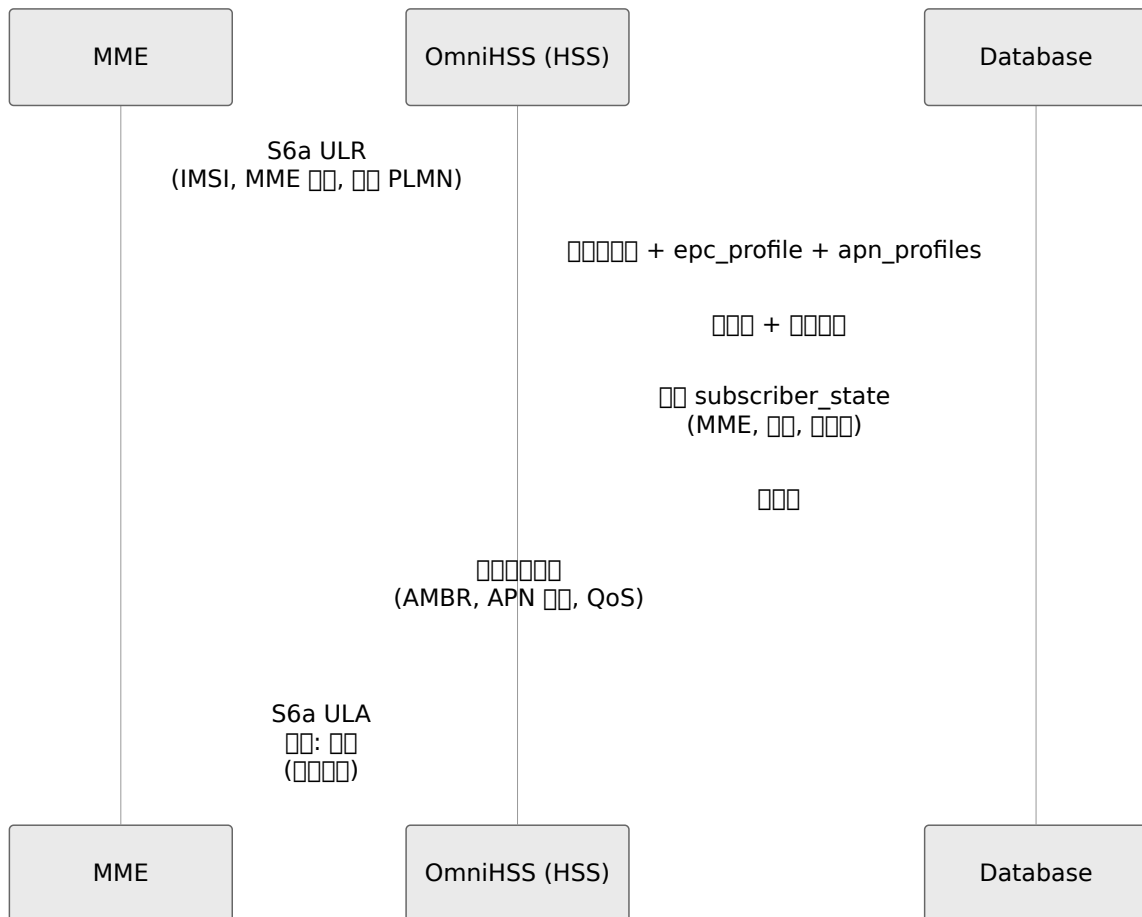


AVP:

- : (IMSI), -PLMN-Id,)
- : - (RAND, AUTN, XRES, KASME)

(ULR/ULA)

MME HSS)

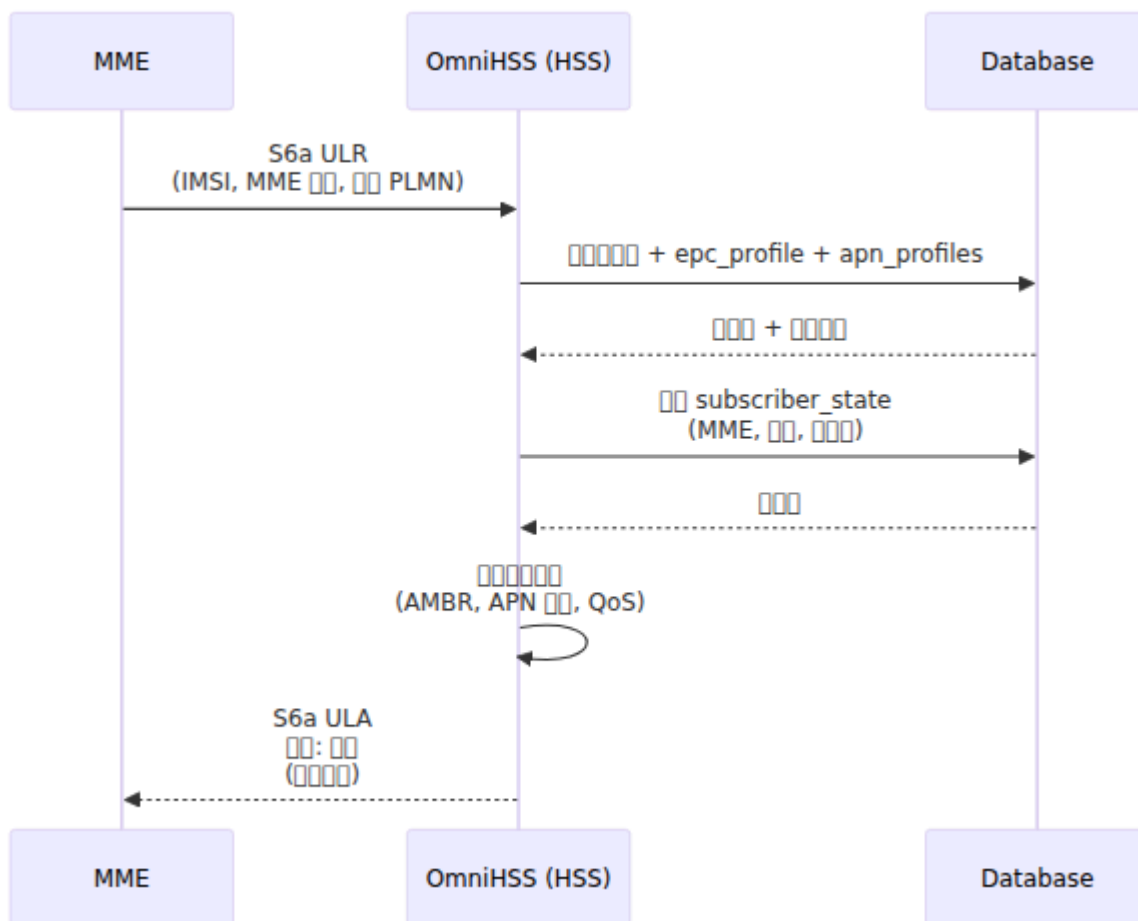


AVP:

- IMSI: IMSI (IMSI), RAT type, ULR type, PLMN-Id, UE-SRVCC-Capability
- AMBR: AMBR (AMBR, APN-ID, QoS-ID)

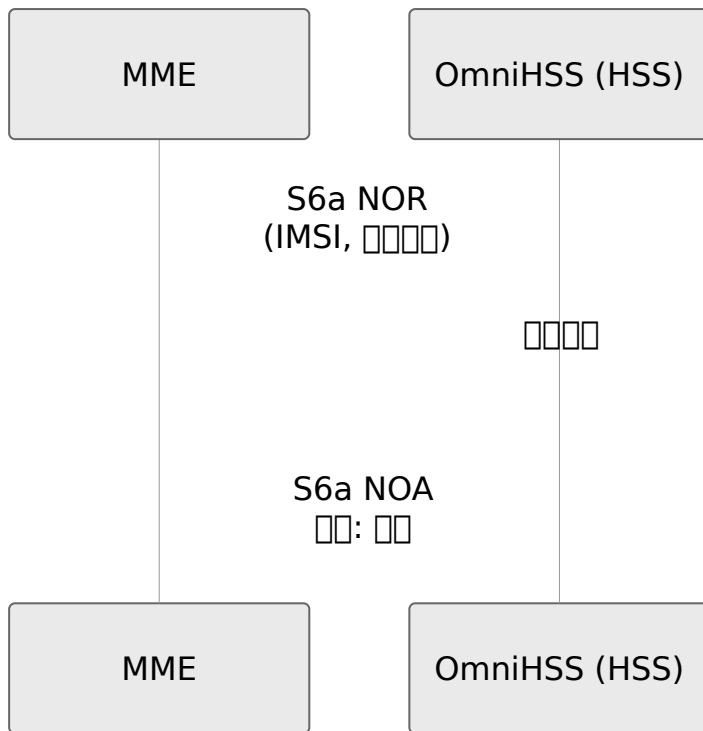
UE ID (PUR/PUA)

MME ID HSS ID



Sequence (NOR/NOA)

MME ID HSS ID

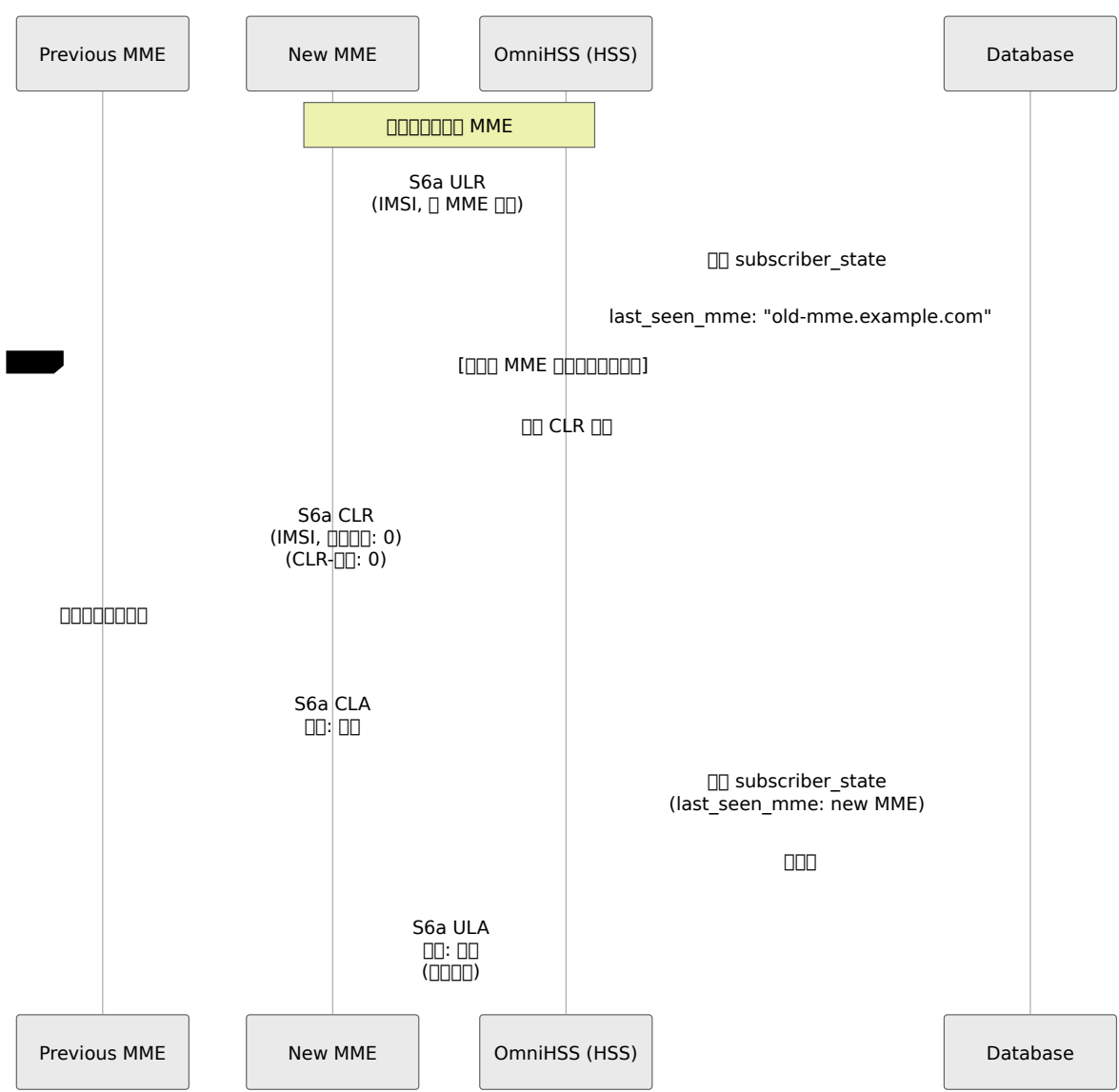


IMSI (CLR/CLA)

HSS IMSI IMSI MME IMSI IMSI OmniHSS IMSI CLR IMSI

IMSI **CLR (MME IMSI)**

IMSI MME IMSI IMSI OmniHSS IMSI IMSI MME IMSI CLR IMSI IMSI IMSI

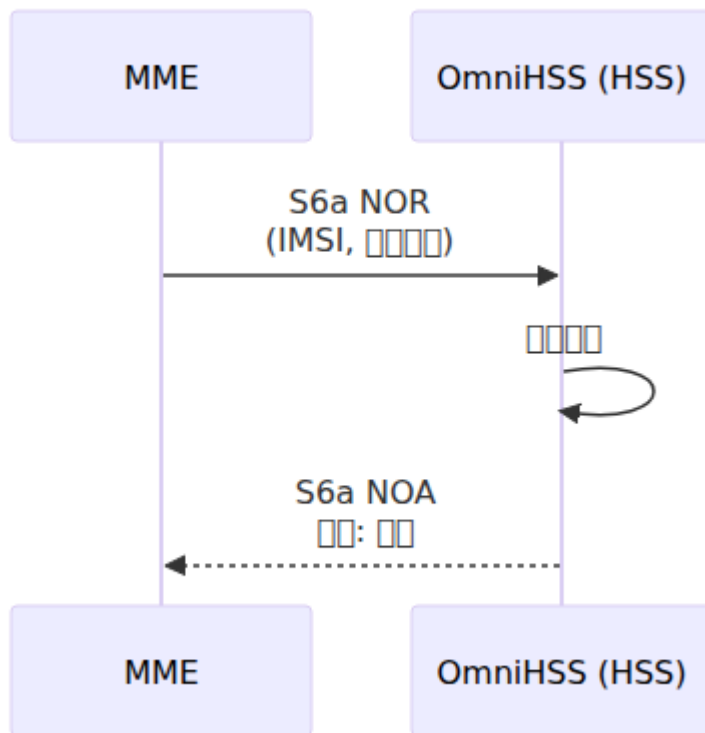


AVP (CLR):

- IMSI: IMSI
- MME ID: MME ID
- MME ID: MME ID
- MME ID: 0 (MME ID)
- CLR-ID: 0
- IMSI: IMSI

CLR (API ID)

API ID CLR



AVP (CLR):

- IMSI: IMSI
- MME: MME
- MME: MME
- :subscription_withdrawal (3GPP TS 29.272)
- CLR-:
 - s6a_indicator: 1 (S6a)
 - reattach_required: 1 (UE)

OmniHSS 3GPP TS 29.272

項目	値	項目	項目
MME 番号	0	MM MME 番号	MM MME 番号 ULR 番号
SGSN 番号	1	SGSN 番号	3G/2G 番号
番号	2	番号	API 番号
番号 IWF	3	番号	番号
番号	4	番号	番号

CLR-番号

CLR-番号 AVP 番号

項目	値	項目
S6a/S6d 番号	0	1 = S6a 番号
番号	1	1 = UE 番号

CLR-番号:

```
clr_flags: %{
  s6a_indicator: 1,          # S6a 番号
  reattach_required: 1      # 番号
}
```

IMSI 番号

OmniHSS 番号 (IMSI) MME 番号 MSISDN番号 IMSI 番号 CLR 番号

1: MSISDN番号 IMSI

□□□ A:

- ```
- IMSI: 999000123456789
- MSISDNs: ["+1234567890", "+9876543210"]
- last_seen_mme: "mme01.operator.com"
```

XXXXXXXX MME XX

- 消息 **CLR** 有 "mme01.operator.com" IMSI 有 999000123456789
- 消息 MSISDN 消息内容 SIM
- 消息 AVP 消息 IMSI 消息 MSISDNs

## 2: IMSI and MSISDN

OmniHSS 0000 00 **MSISDN** 000000 MSISDN 00000000000000000000/000000

□□□ A:

- ```
- IMSI: 999000111111111
- MSISDN: "+1234567890"
- last_seen_mme: "mme01.operator.com"
```

□□□ B□□□□□:

- ```
- IMSI: 999000222222222
- MSISDN: "+1234567890" # [] MSISDN[] SIM/IMSI
- last_seen_mme: "mme02.operator.com"
```

□□□□ B □□□□

- **CLR** IMSI =
- A mme01
- B mme02
- 

**3: CLR MSISDN**

111

- **CLR** last\_seen\_mme
- **MSISDN** IMSI

- IMSI 000 MME 000000

0000

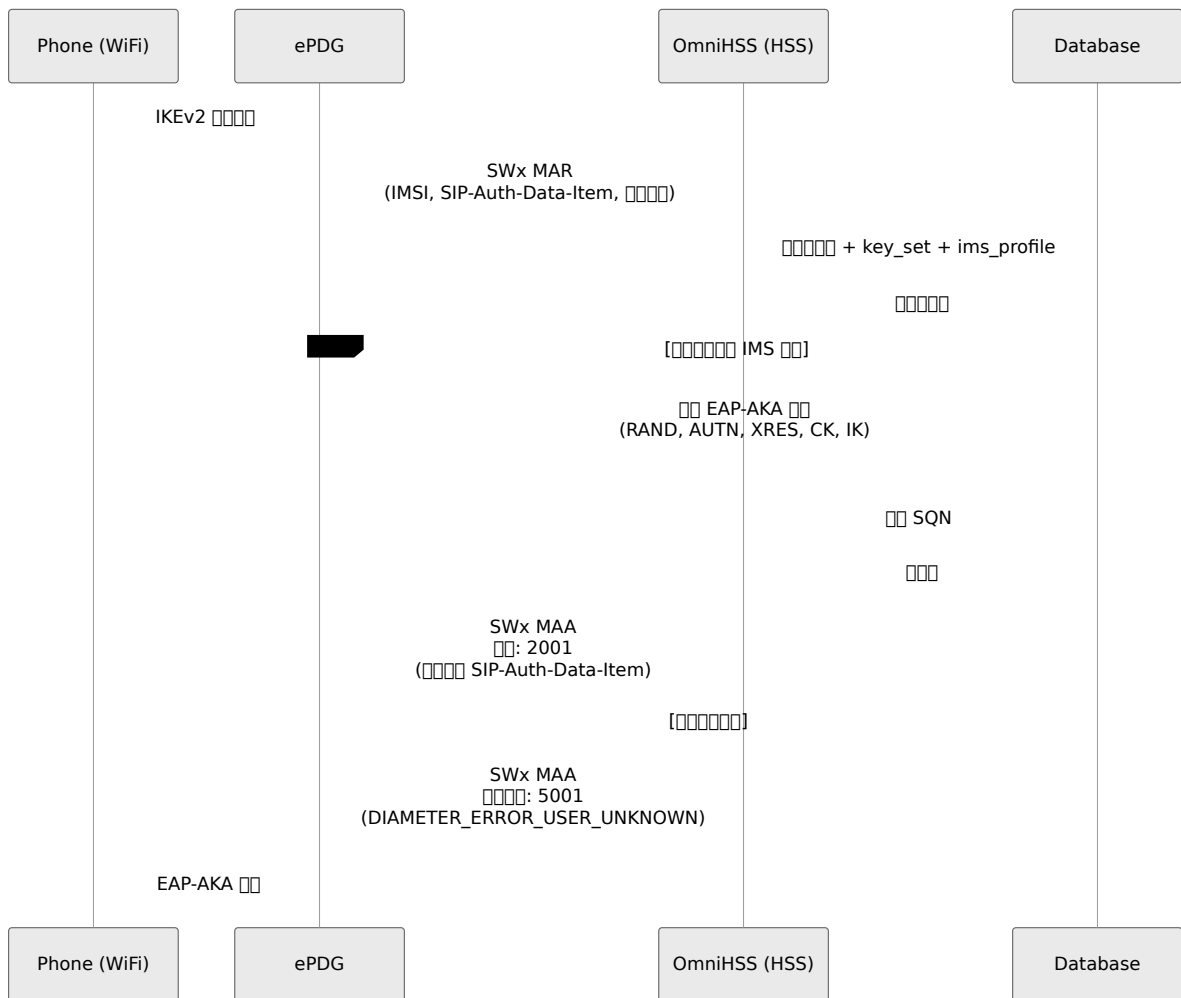
1. **IMSI** 000: CLR 000000 0 **IMSI**000000 MSISDN0 subscriber\_state 000000 (IMSI) 00 last\_seen\_mme0
2. 0000: 00000000000000 MME 000000 CLR 000000000000000000
3. 00000000 **MME** 0000 **CLR:** 00 last\_seen\_mme 0 nil00000000000000 ULR 0000 0 CLR0
4. 0000000: 00 CLR00 ULR 0000000000 Subscription-Data AVP000000 MME 000000 0000
5. 00: CLR 0000000000000000 MME 0 ULA 000000 MME 0 CLA0
6. **CLA** 00: OmniHSS 00 CLA 000000000000000000000000000000 HSS 0000

## SWx 00 (0 3GPP 00 / WiFi 00)

SWx 000000 ePDG00000000000000 HSS000000 3GPP 000000000000000000000000 WiFi 00 00

### 00000000 (MAR/MAA)

ePDG 0000 WiFi 000000 EAP-AKA 000000



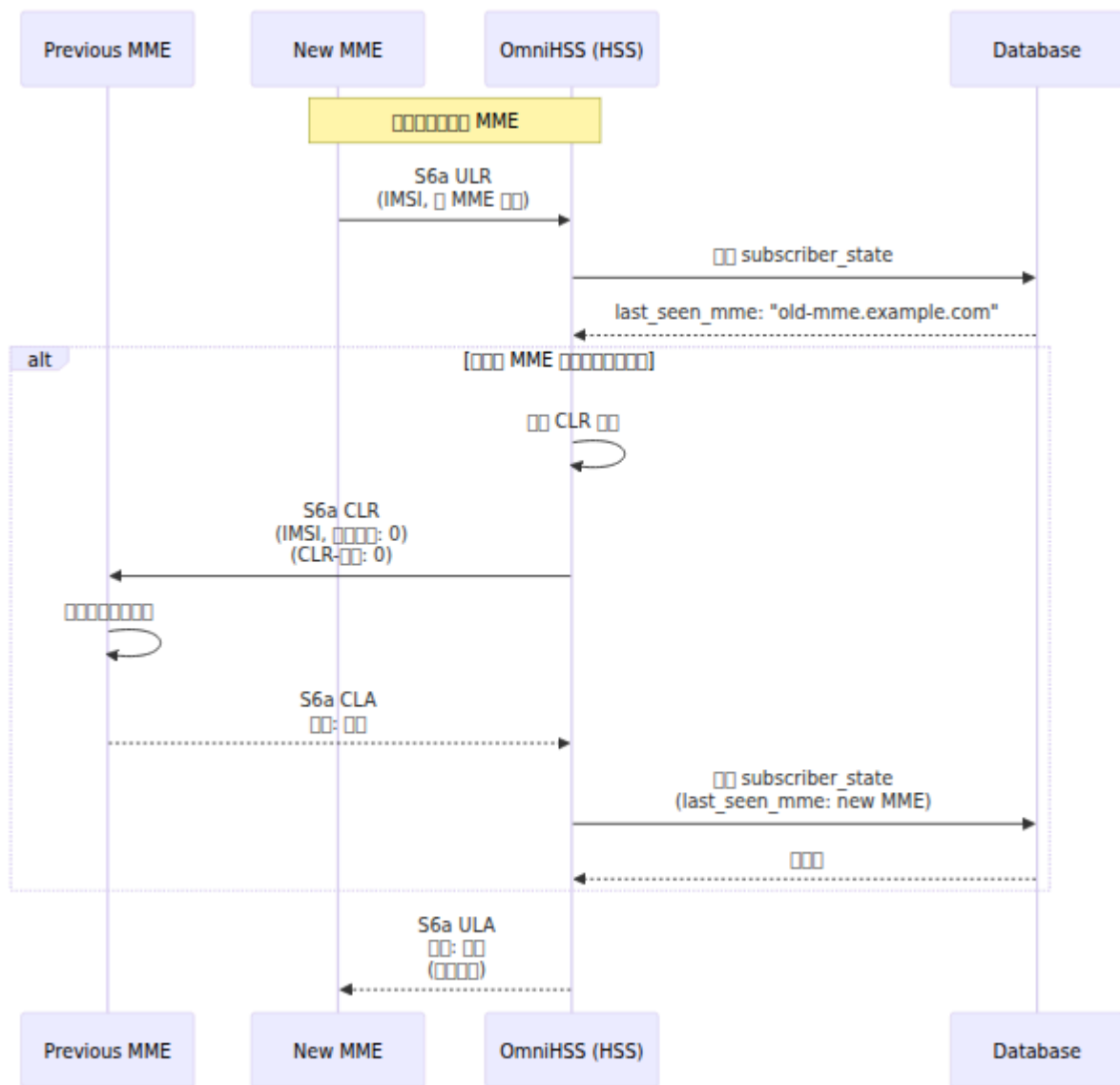
## AVP:

- : (IMSI @ IMSI@realm), SIP-Auth-Data-Item ( ), SIP-Number-Auth-Items
- : SIP-Auth-Data-Item (SIP-Authenticate, SIP-Authorization, , )

**SQN** : UE SQN ePDG SIP-Authorization AVP OmniHSS AUTS 3GPP TS 33.102 IND SQN

## (SAR/SAA)

ePDG WiFi



□□□□□□□□:



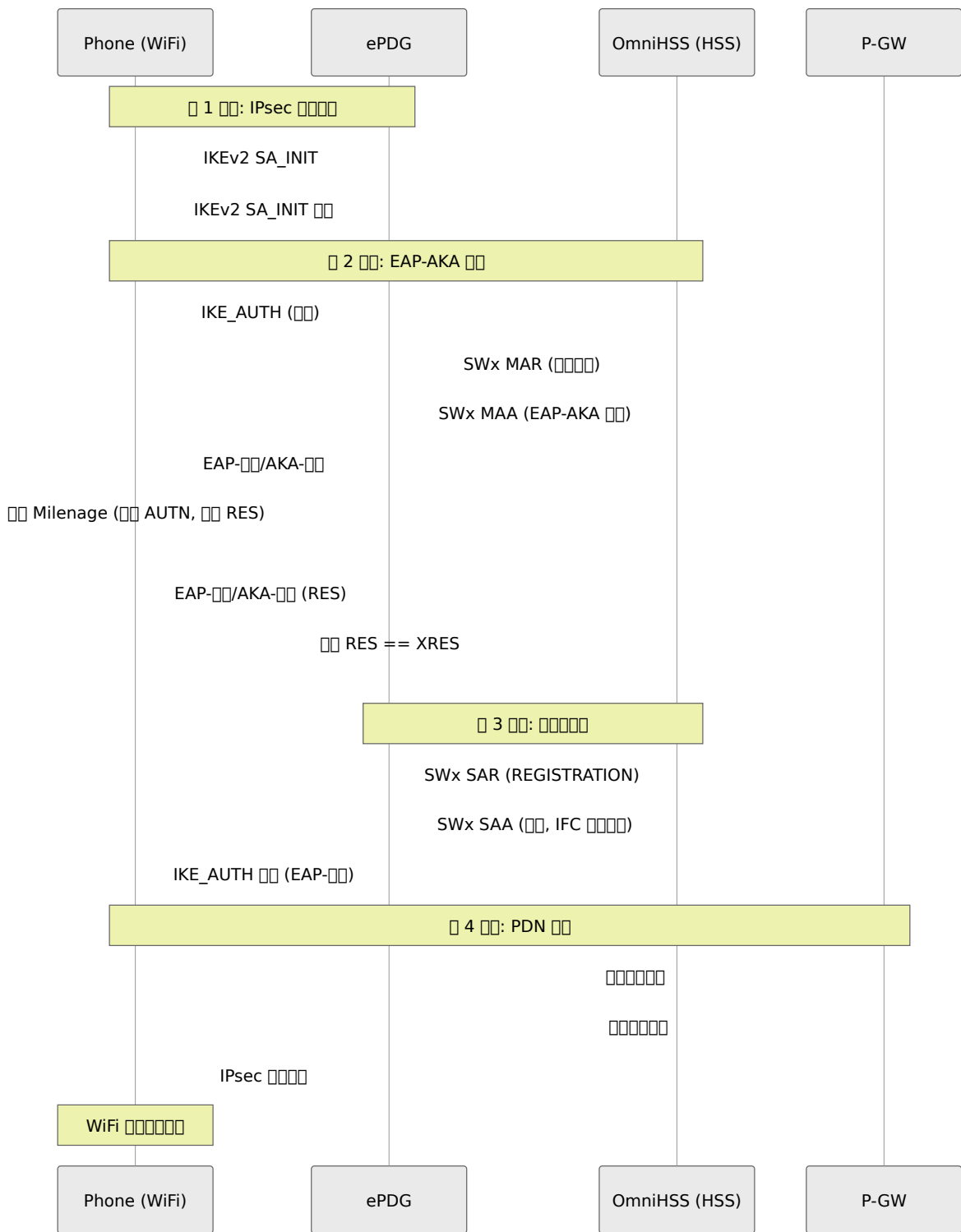
| 코드                            | 번호 | 설명          |
|-------------------------------|----|-------------|
| NO_ASSIGNMENT                 | 0  | 할당되지 않음     |
| REGISTRATION                  | 1  | WiFi 등록     |
| RE_REGISTRATION               | 2  | 재등록         |
| UNREGISTERED_USER             | 3  | 등록되지 않은 사용자 |
| TIMEOUT_DEREGISTRATION        | 4  | 등록 시간 초과    |
| USER_DEREGISTRATION           | 5  | 사용자 등록 해제   |
| AUTHENTICATION_FAILURE        | 7  | 인증 실패       |
| ADMINISTRATIVE_DEREGISTRATION | 8  | 관리자 등록 해제   |

AVP:

- 코드: IMSI, 전화번호, ePDG 주소, APN
- 코드: IFC (XML), 3GPP 코드

WiFi

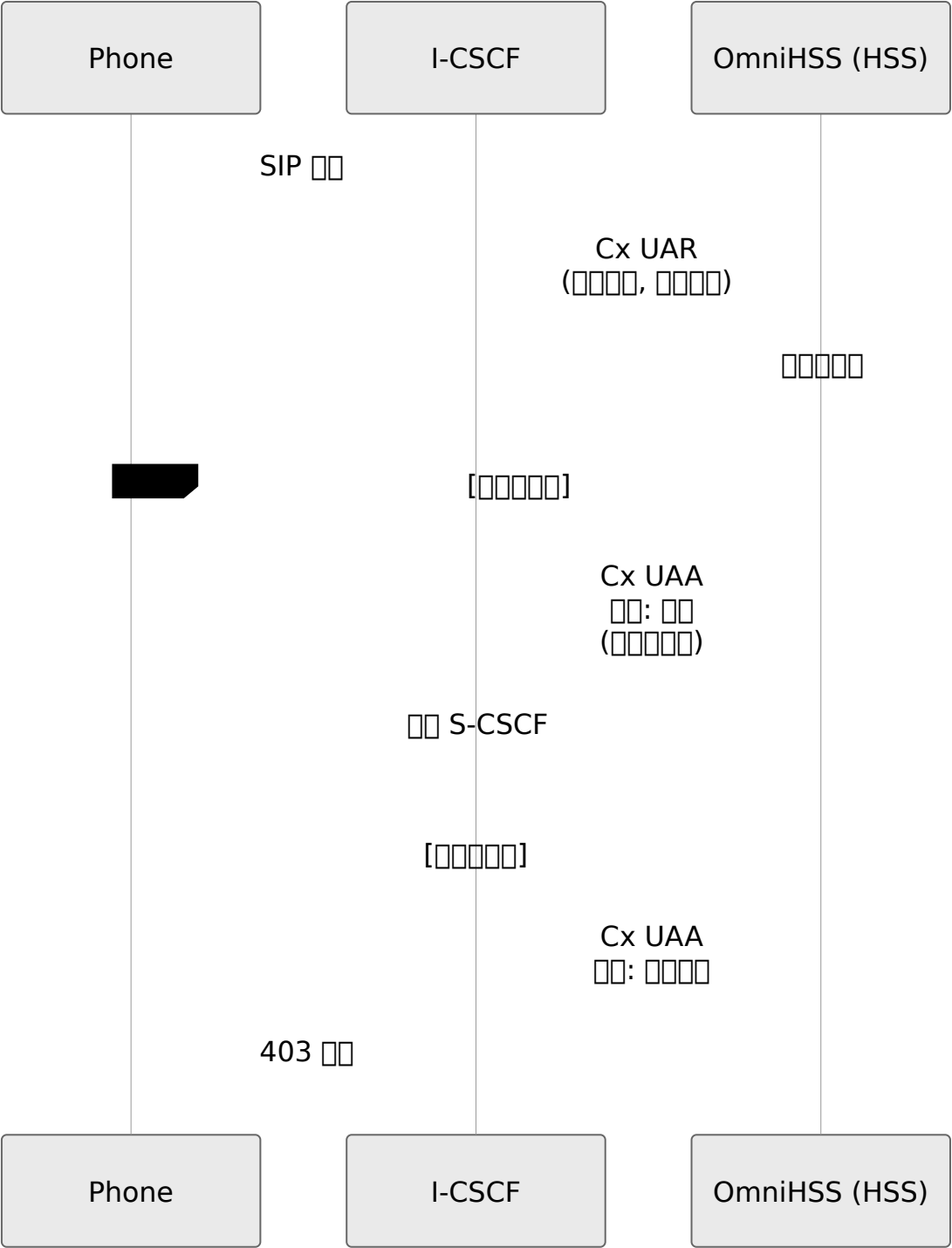
WiFi



# Cx 消息 (IMS)

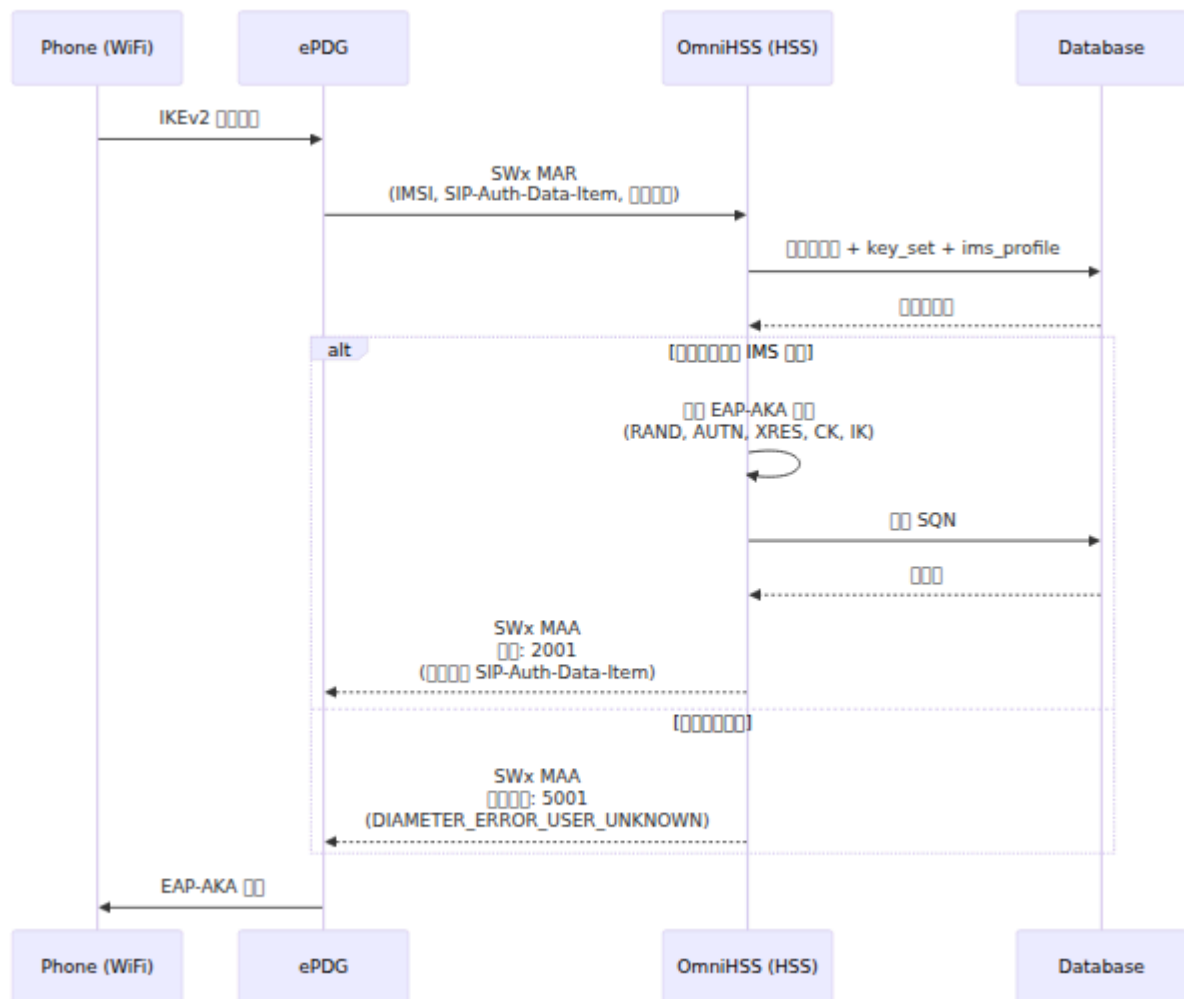
## 注册消息 (UAR/UAA)

I-CSCF 注册消息



## 鉴权鉴权 (SAR/SAA)

S-CSCF 向/从HSS IMS 鉴权

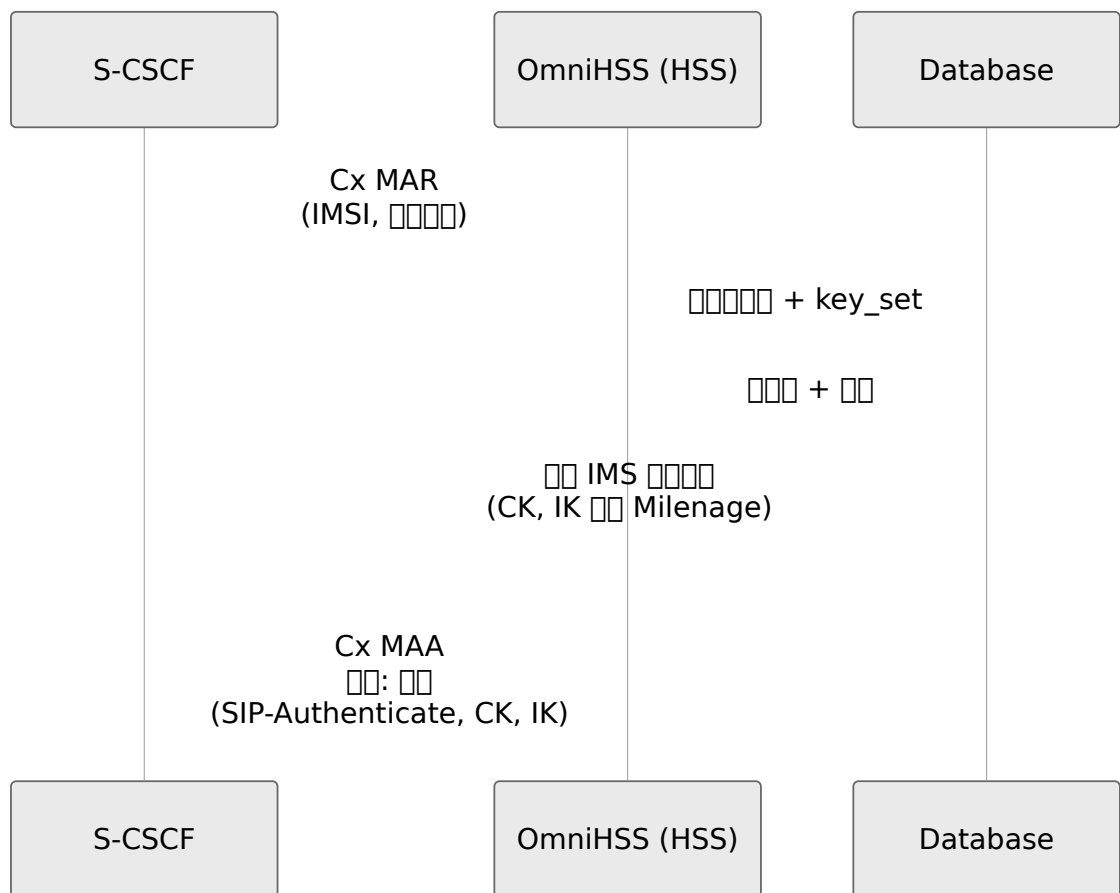


IFC 鉴权:

- `{{imsi}}` → IMSI
- `{{msisdns}}` → 鉴权
- `{{mcc}}`, `{{mnc}}` → PLMN 鉴权

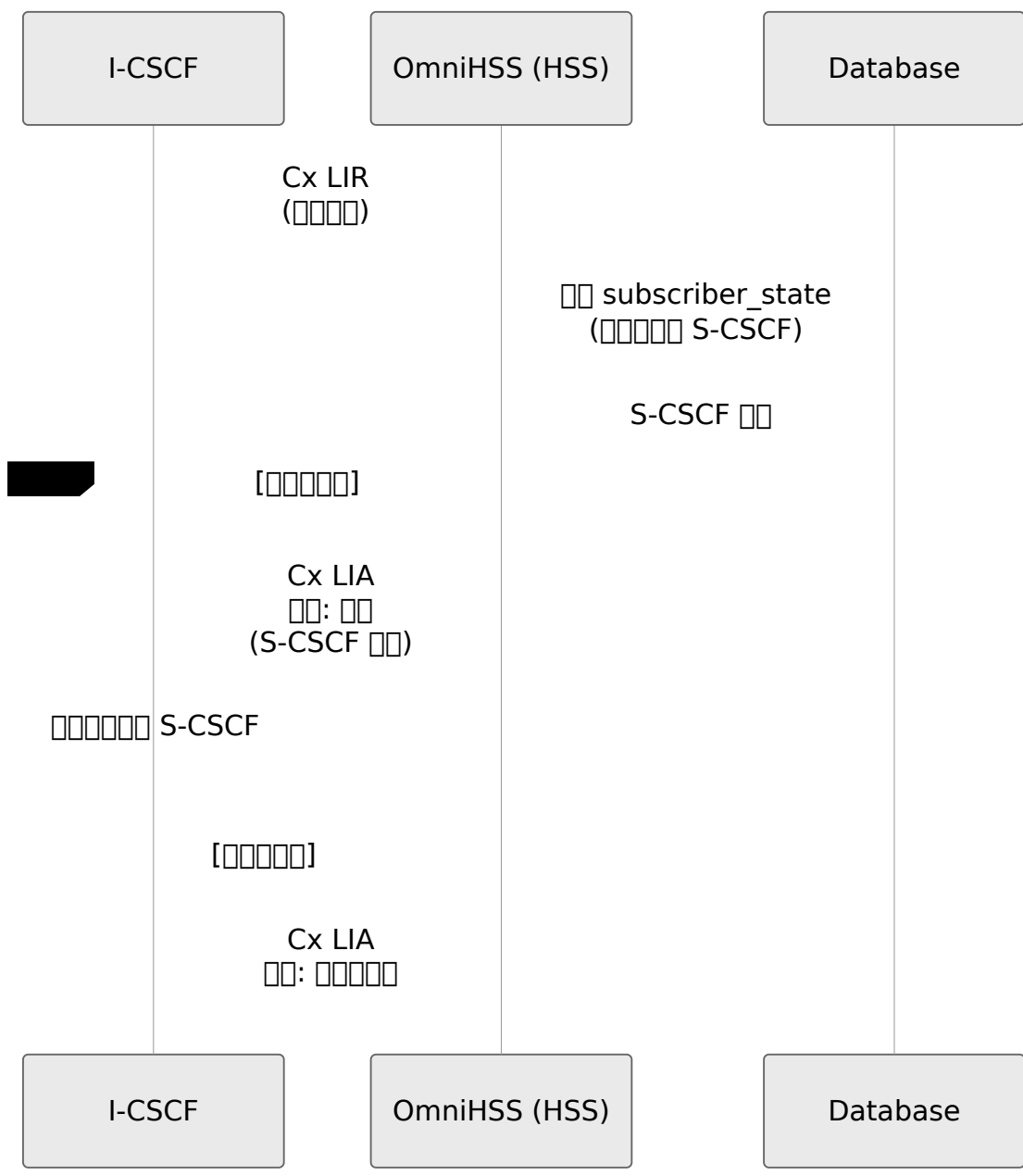
## 鉴权鉴权 (MAR/MAA)

S-CSCF 向 IMS 鉴权



## IMS (LIR/LIA)

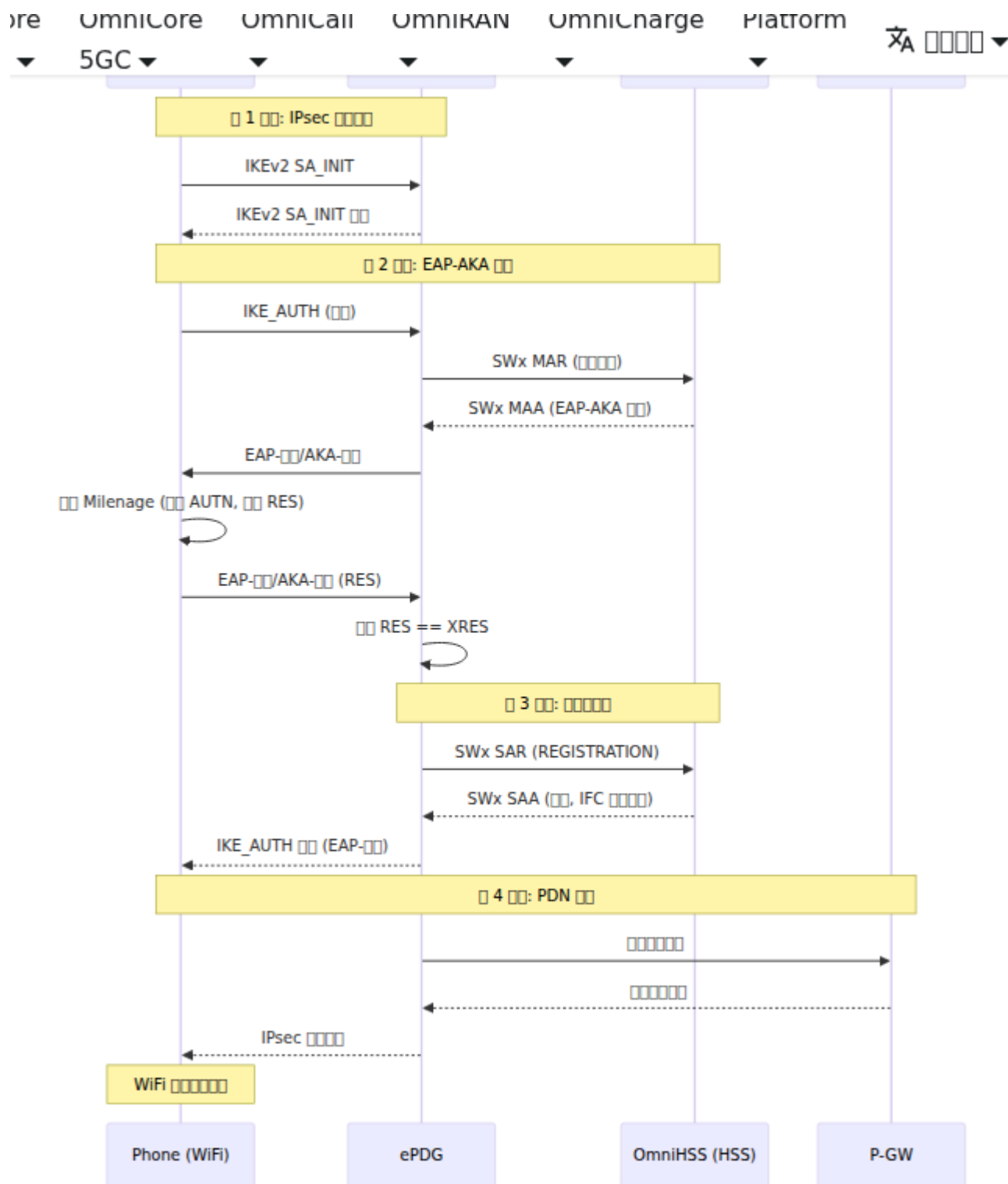
I-CSCF 通过 S-CSCF 注册



# Sh (IMS )

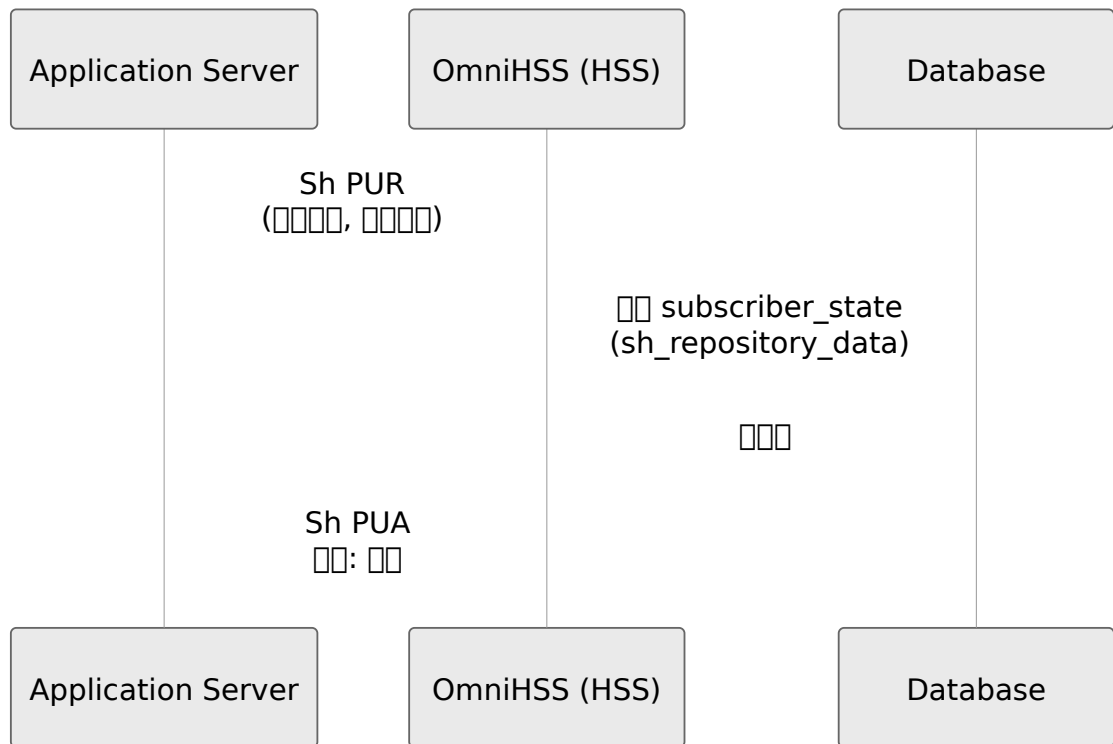
Sh RepositoryData ( ) / MMTEL / IMS-ODB  
REST API

## (UDR/UDA)



## Registration (PUR/PUA)

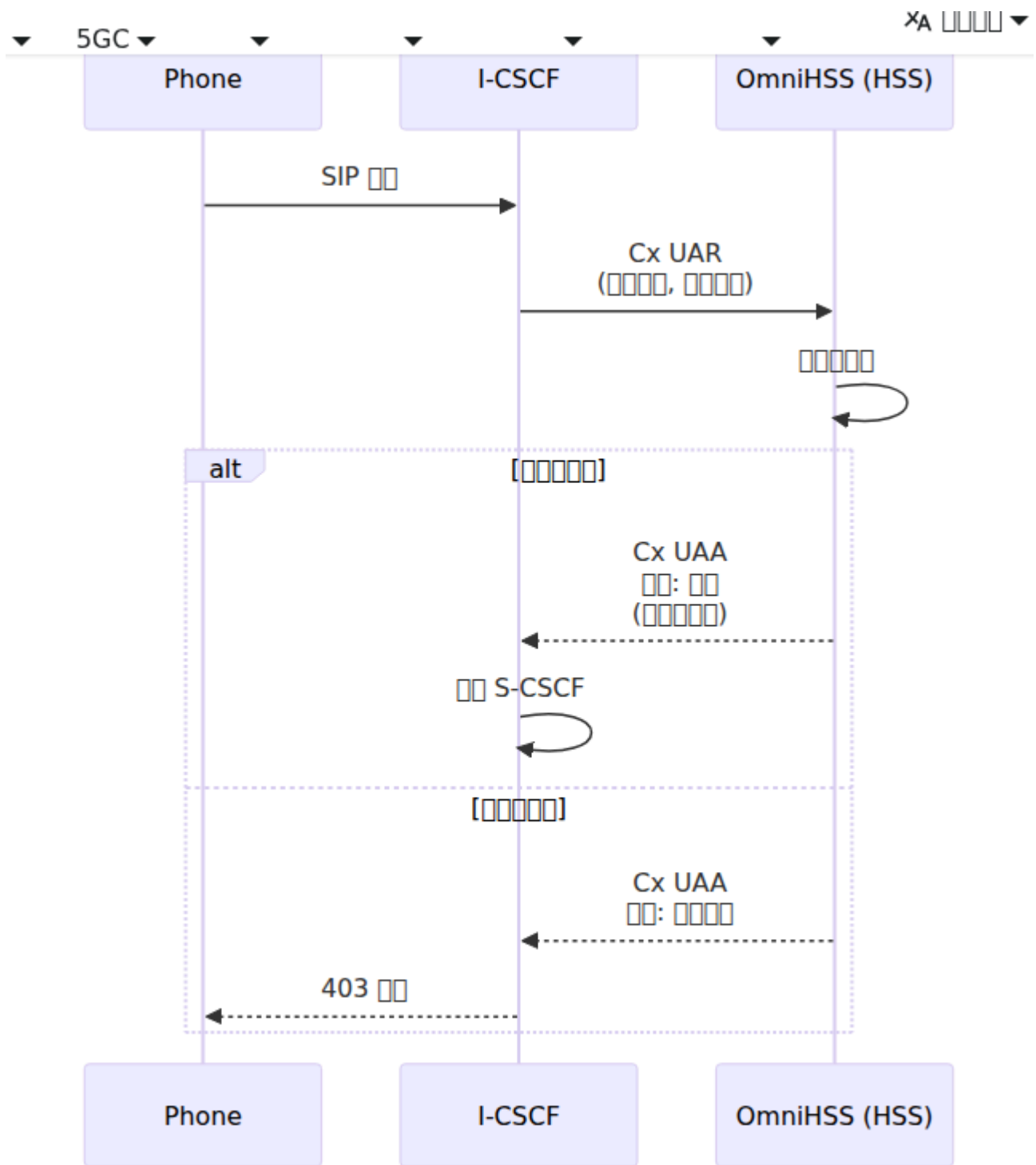
Registration (PUR/PUA)



## Subscriber State (SNR/SNA)

Subscriber State (SNR/SNA)





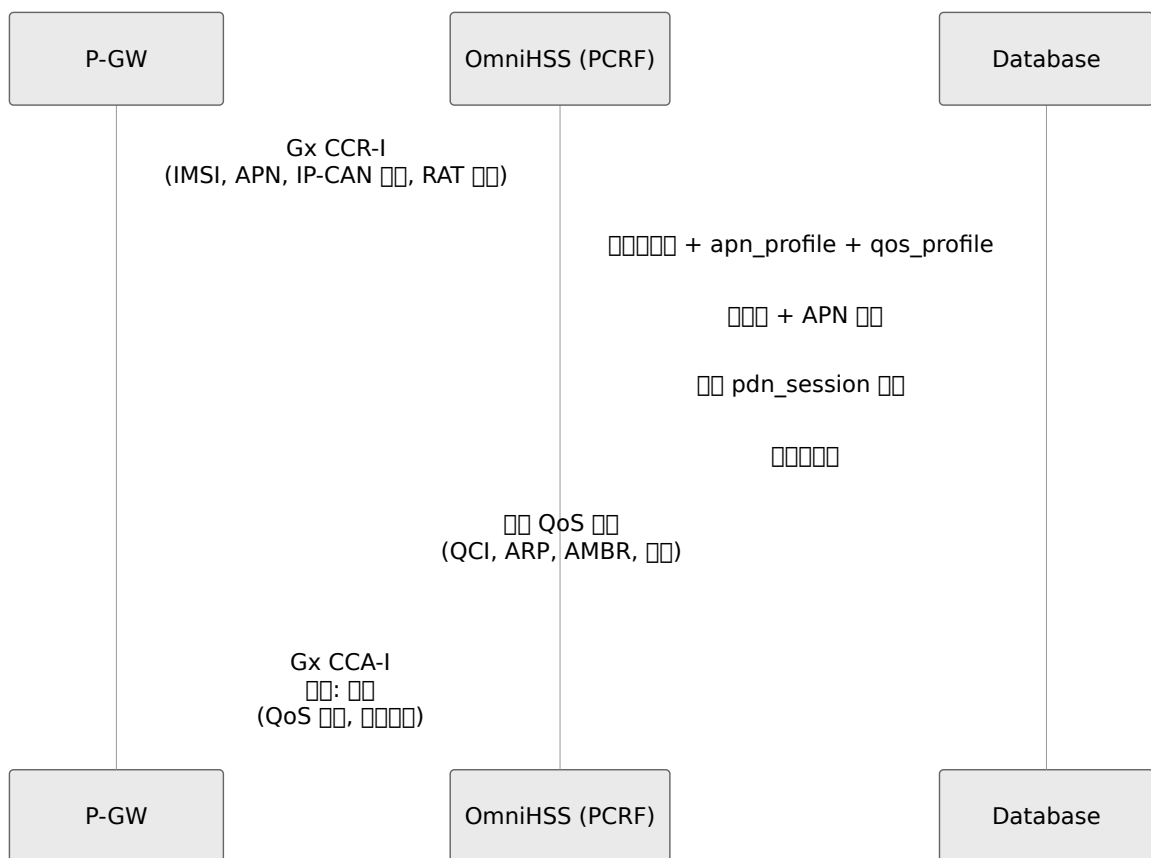
## Gx ( )

OmniHSS Gx PCRF

PCRF QoS

## ❖❖❖❖❖❖ - ❖❖ (CCR-I/CCA-I)

P-GW ❖ PDN ❖❖❖❖❖❖❖❖❖❖❖❖

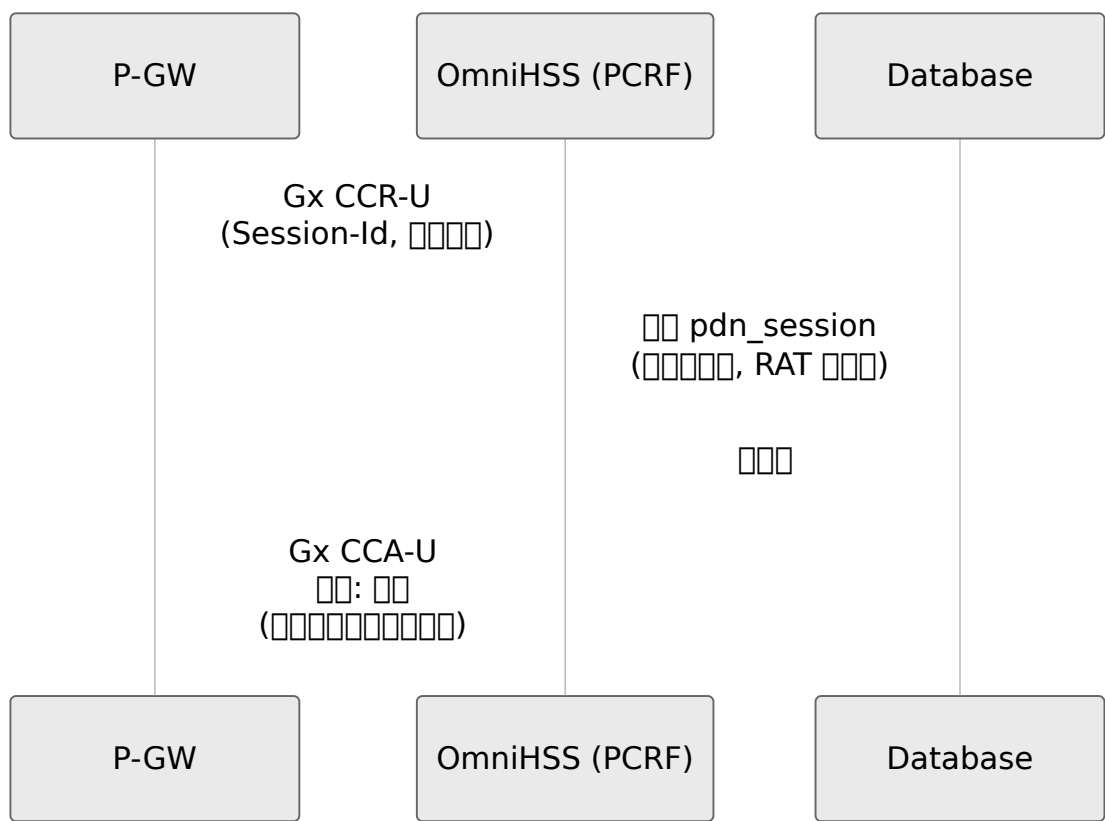


❖❖ **AVP:**

- ❖❖: ❖❖ ID (IMSI), ❖❖❖❖ ID (APN), RAT ❖❖, IP-CAN ❖❖
- ❖❖: QoS ❖❖ (QCI, ARP, AMBR), ❖❖❖❖❖❖

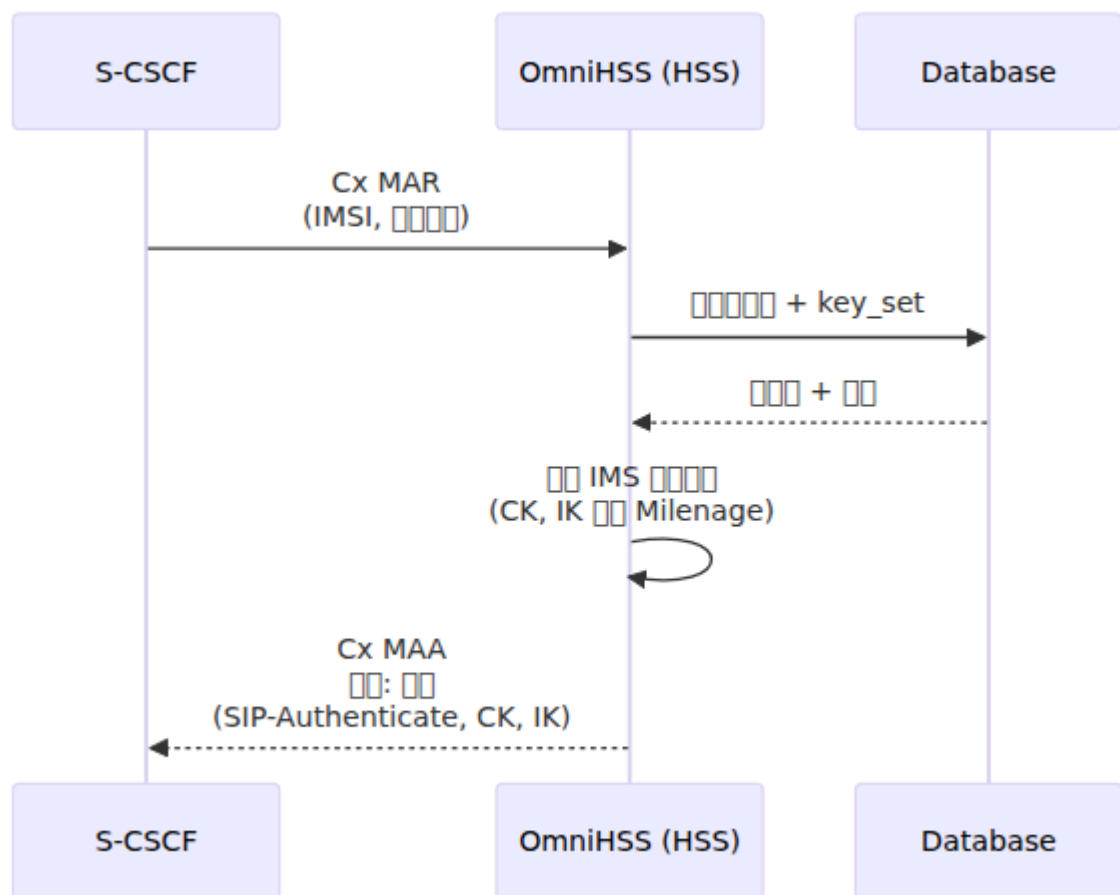
## ❖❖❖❖❖❖❖❖ - ❖❖ (CCR-U/CCA-U)

P-GW ❖❖❖❖❖❖❖❖❖❖❖❖



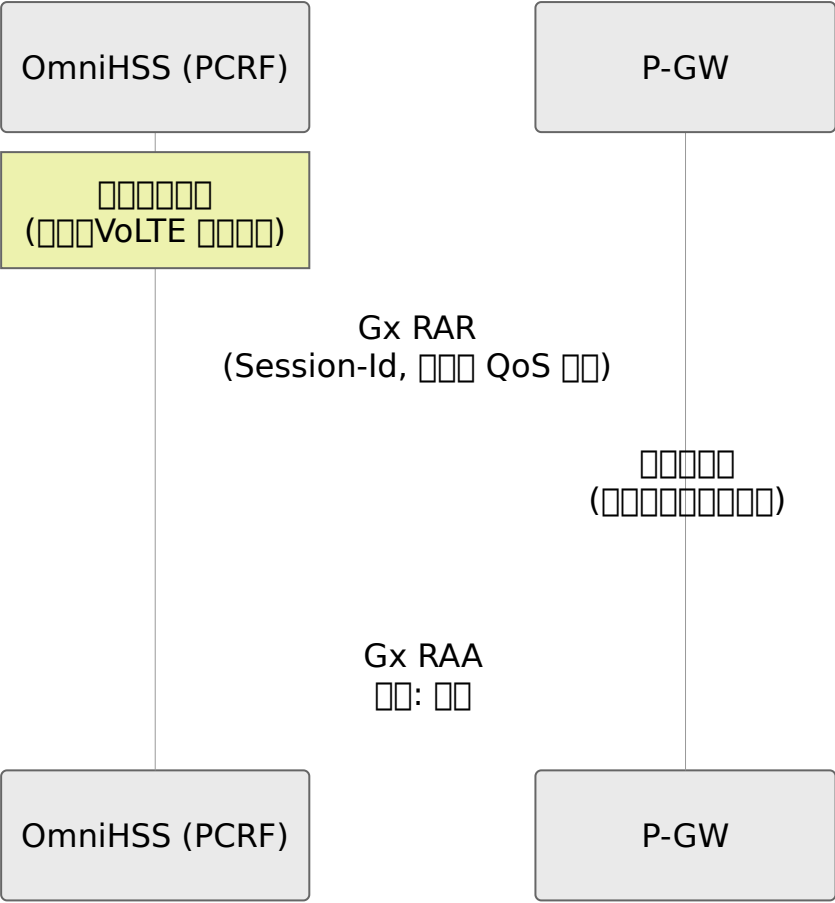
# 0000000 - 00 (CCR-T/CCA-T)

P-GW 00 PDN 000000



## [IMSI] IMS [IMSI] (RAR/RAA)

OmniHSS (PCRF) [IMSI] P-GW [IMSI]



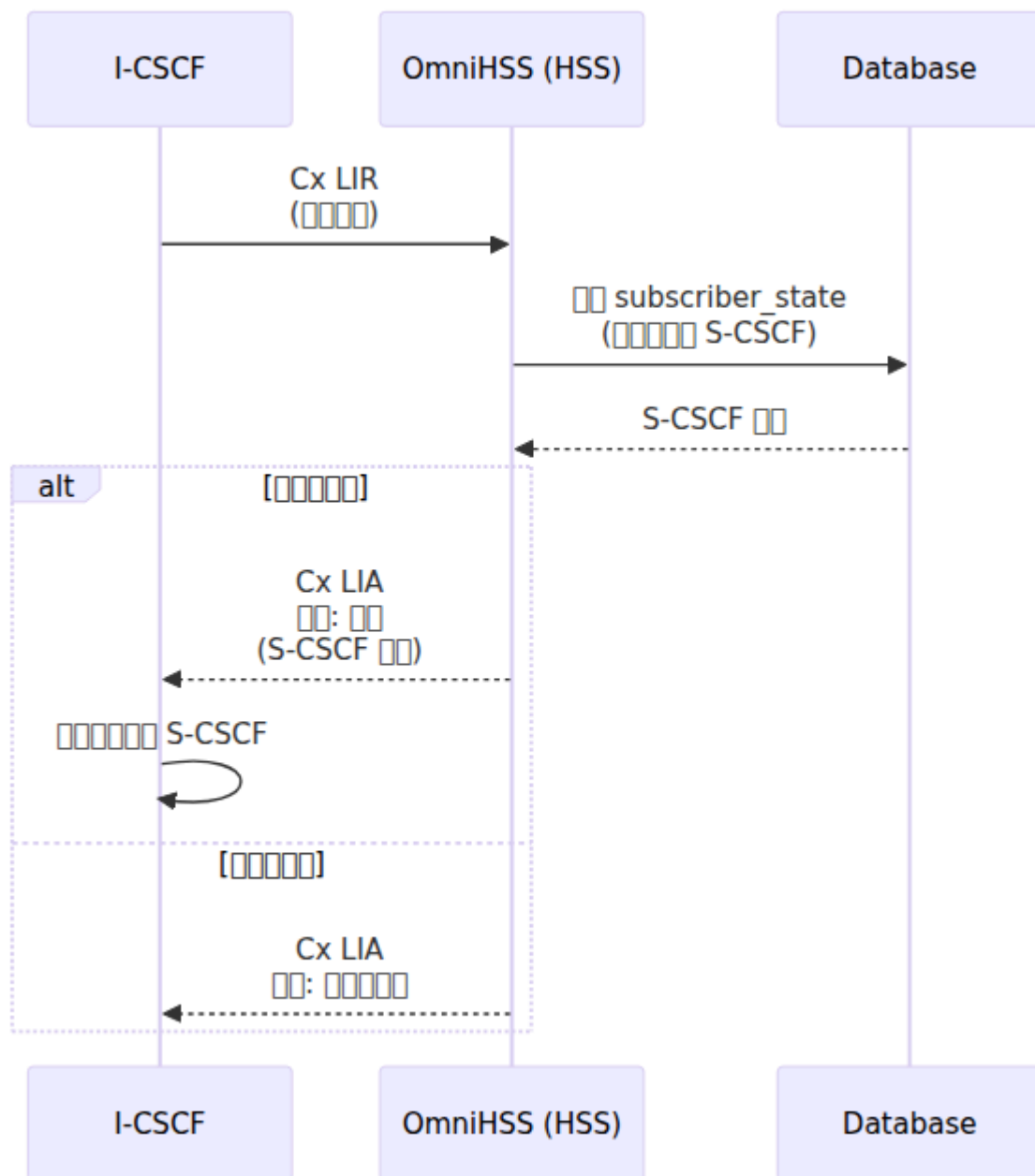
## Rx (IMS)

OmniHSS Rx PCRF IMS

PCRF VoLTE

## AA (AAR/AAA)

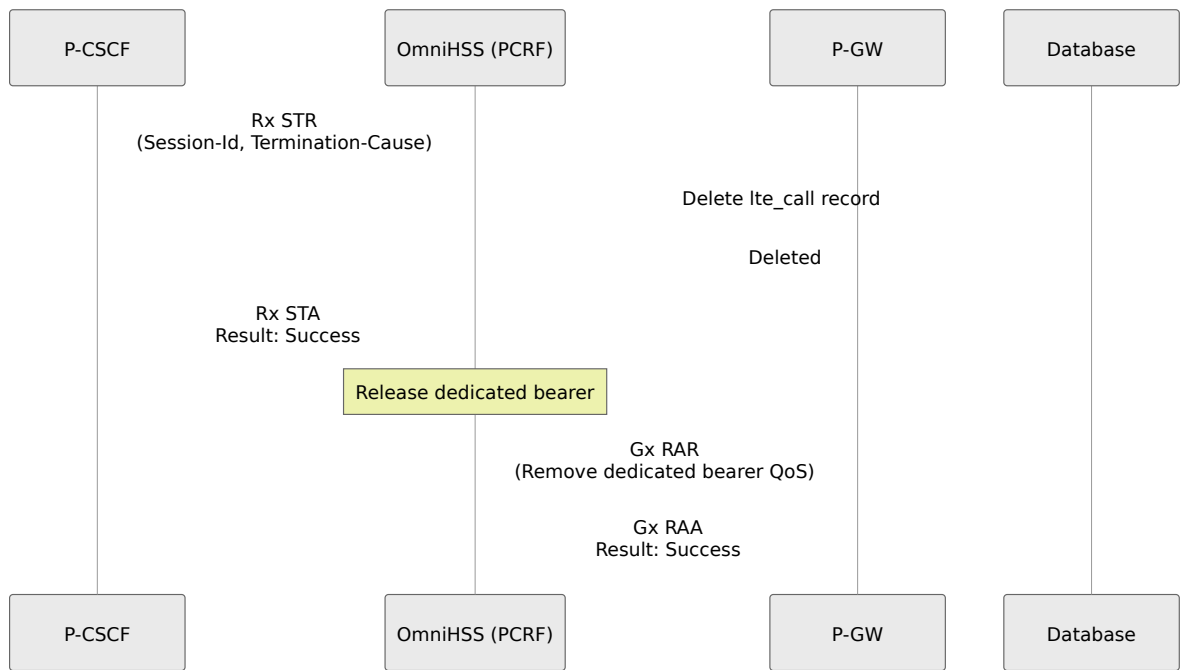
P-CSCF IMS



- SDP
- ( )
- SDF
- Gx RAR

## (STR/STA)

P-CSCF IMS



## Sy ( )

OmniHSS PCRF Sy OCS ( ) Gx Sy ID 16777302 (3GPP TS 29.219)HSS Sy OCS Sy AVP

## (SLR/SLA)

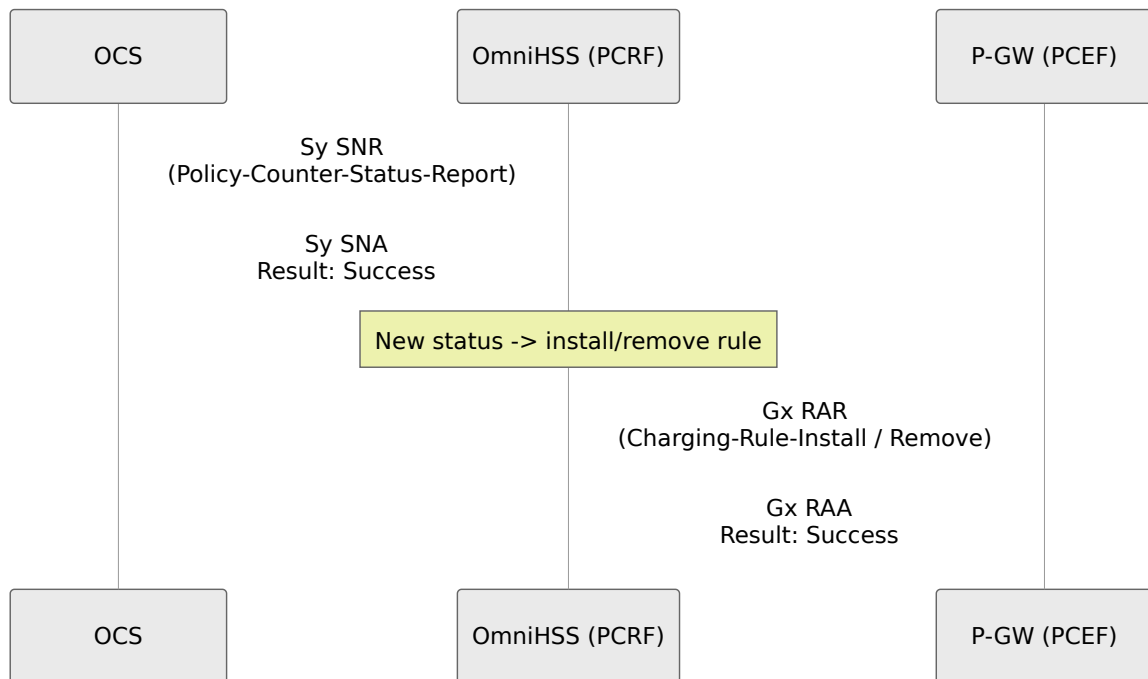
PCRF Sy OCS Gx CCA

Parse error on line 9: ... counter identifier;<br>install/remove -----  
 ---^ Expecting 'SPACE', 'NEWLINE', 'create', 'box', 'end', 'autonumber',  
 'activate', 'deactivate', 'title', 'legacy\_title', 'acc\_title', 'acc\_descr',  
 'acc\_descr\_multiline\_value', 'loop', 'rect', 'opt', 'alt', 'par', 'par\_over', 'critical',  
 'break', 'option', 'and', 'else', 'participant', 'participant\_actor', 'destroy', 'note',  
 'links', 'link', 'properties', 'details', 'ACTOR', got 'INVALID'



## Policy-Counter-Status-Report (SNR/SNA)

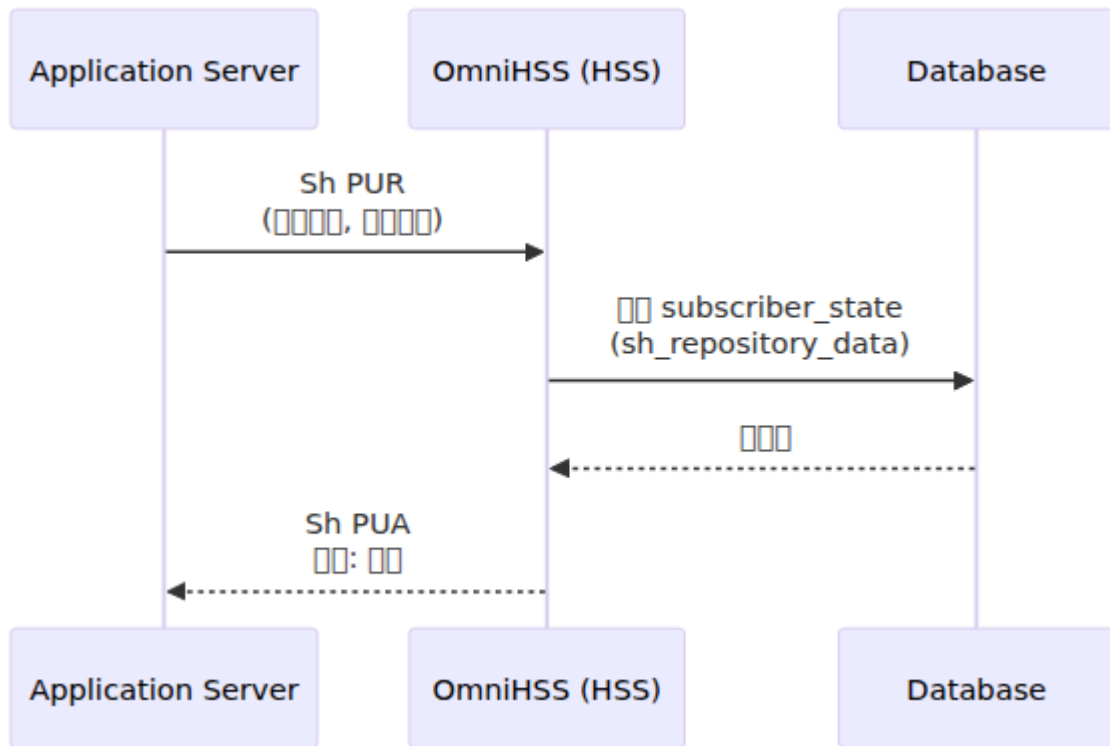
Policy-Counter-Status-Report (SNR/SNA) OCS to PCRF, PCRF to Gx RAR, PGW to PCEF



## Policy-Counter-Status-Report (STR/STA)

Policy-Counter-Status-Report (STR/STA) PCRF to Sy





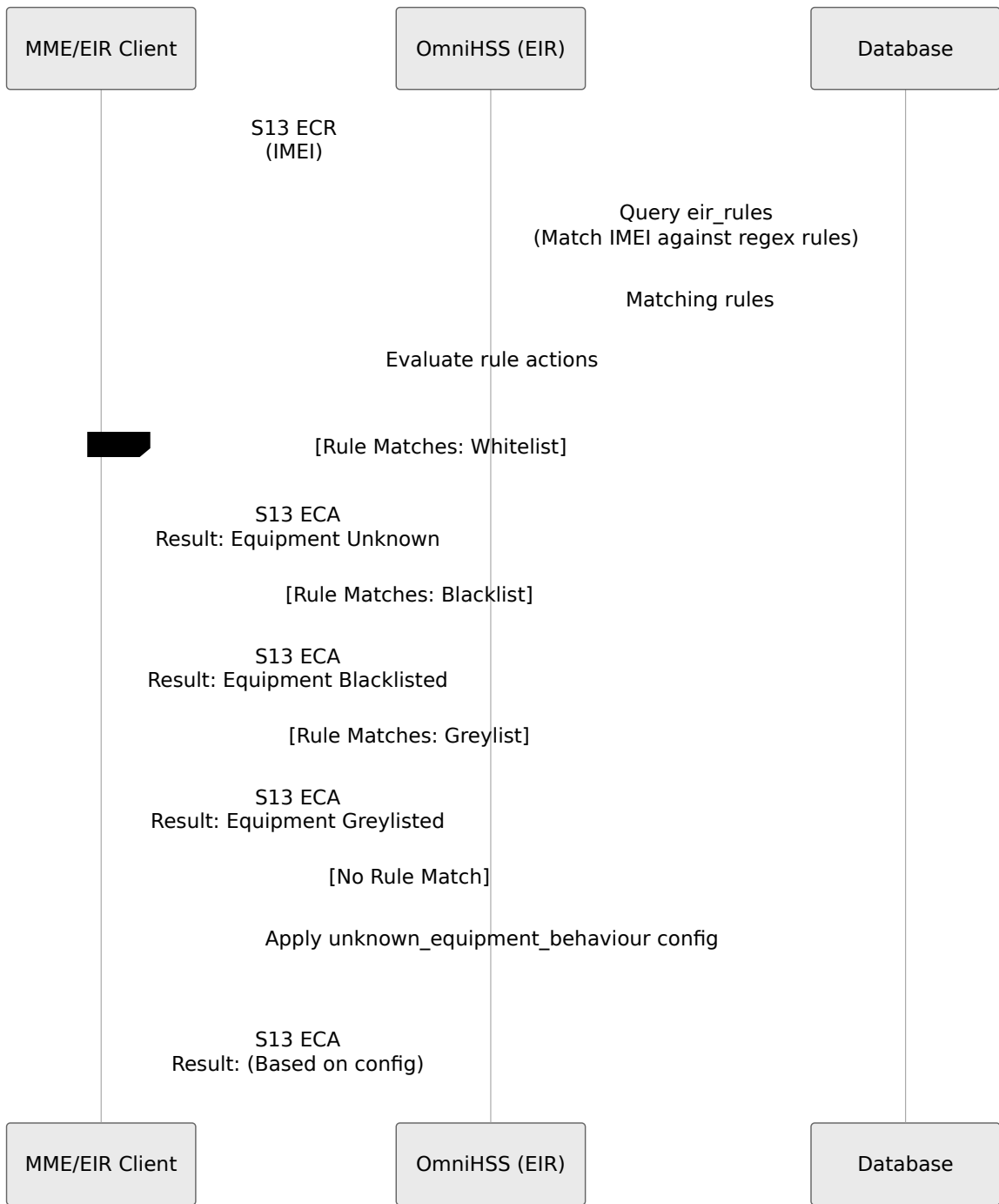
## S13 (EIR)

OmniHSS 00 S13 0000 EIR (000000)0

000 **EIR** 00 00000000000000**IMEI** 0000000000

ME  (ECR/ECA)

00 EIR 00000 MME000000000



□□□□□□

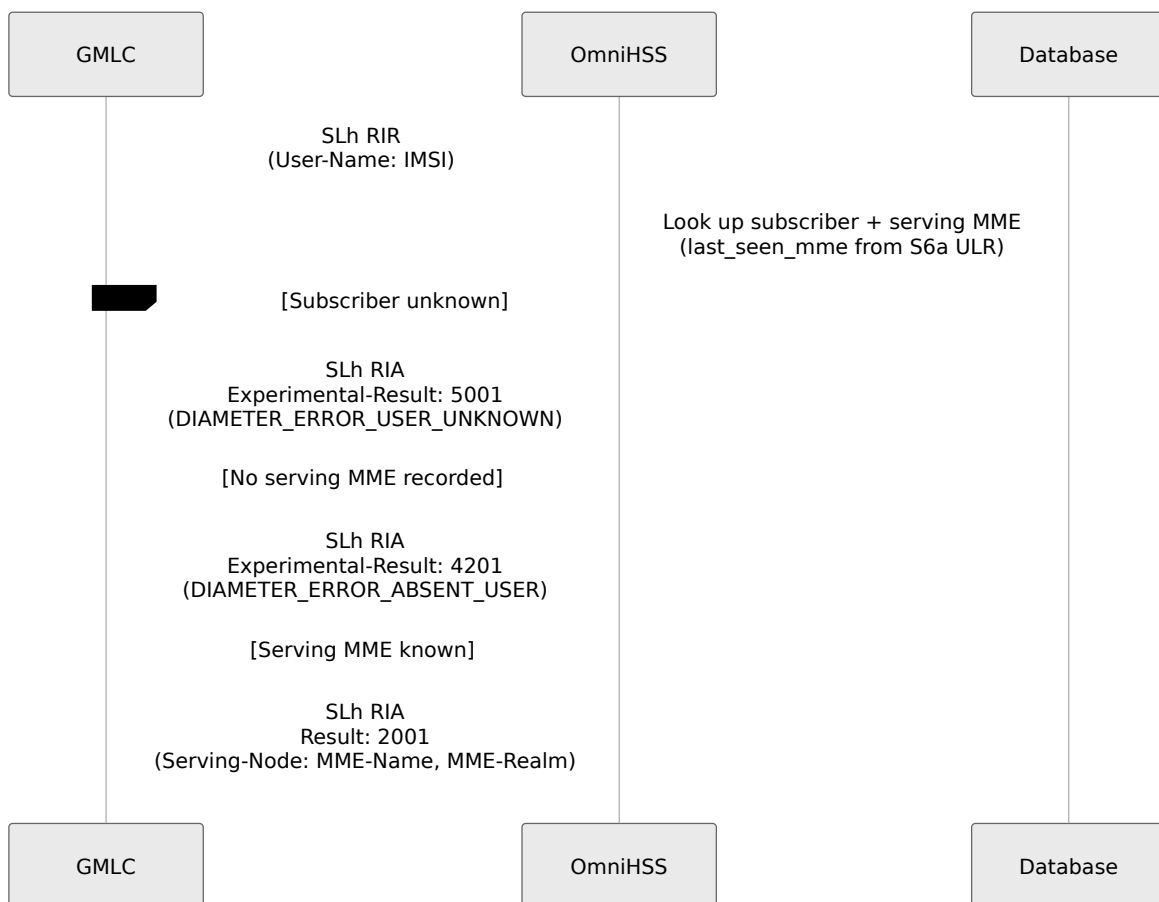
- □□□□ (0) - □□□□□□□□
- □□□□□ (1) - □□□□□
- □□□□□ (2) - □□□□□□□□

# SLh 消息 (LCS / 位置)

OmniHSS 向 SLh 发送 **LCS-Routing-Info-Request** (RIR) 消息给 GMLC 请求位置信息。  
❗❗ 消息 MME 消息 MME 向 S6a 请求位置信息。SLh 消息 ID **16777291** (3GPP TS 29.173)。

## LCS 消息 (RIR/RIA)

GMLC 消息



消息

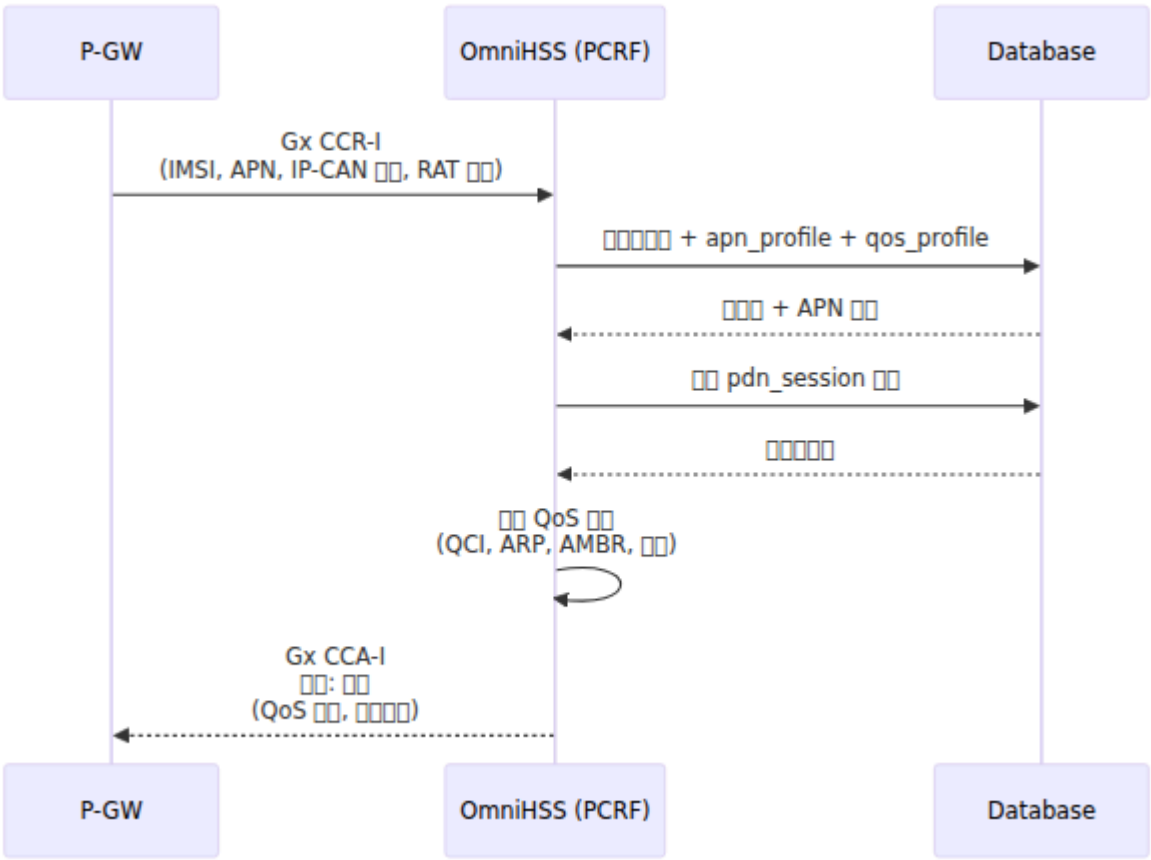
- IMSI 消息 **User-Name** 消息 MSISDN 消息
- 消息 **Serving-Node** 消息 S6a AIR/ULR 消息 **MME-Name**/**MME-Realm** 消息 **subscriber\_state.last\_seen\_mme** / **last\_seen\_realm** 消息 - 消息
- GMLC 消息 RIA 消息 SLg Provide-Location-Request 消息 MME

消息

- **2001** - `Serving-Node`
- **5001** - `DIAMETER_ERROR_USER_UNKNOWN` IMSI/MSISDN
- **4201** - `DIAMETER_ERROR_ABSENT_USER` MME

# VoLTE

VoLTE

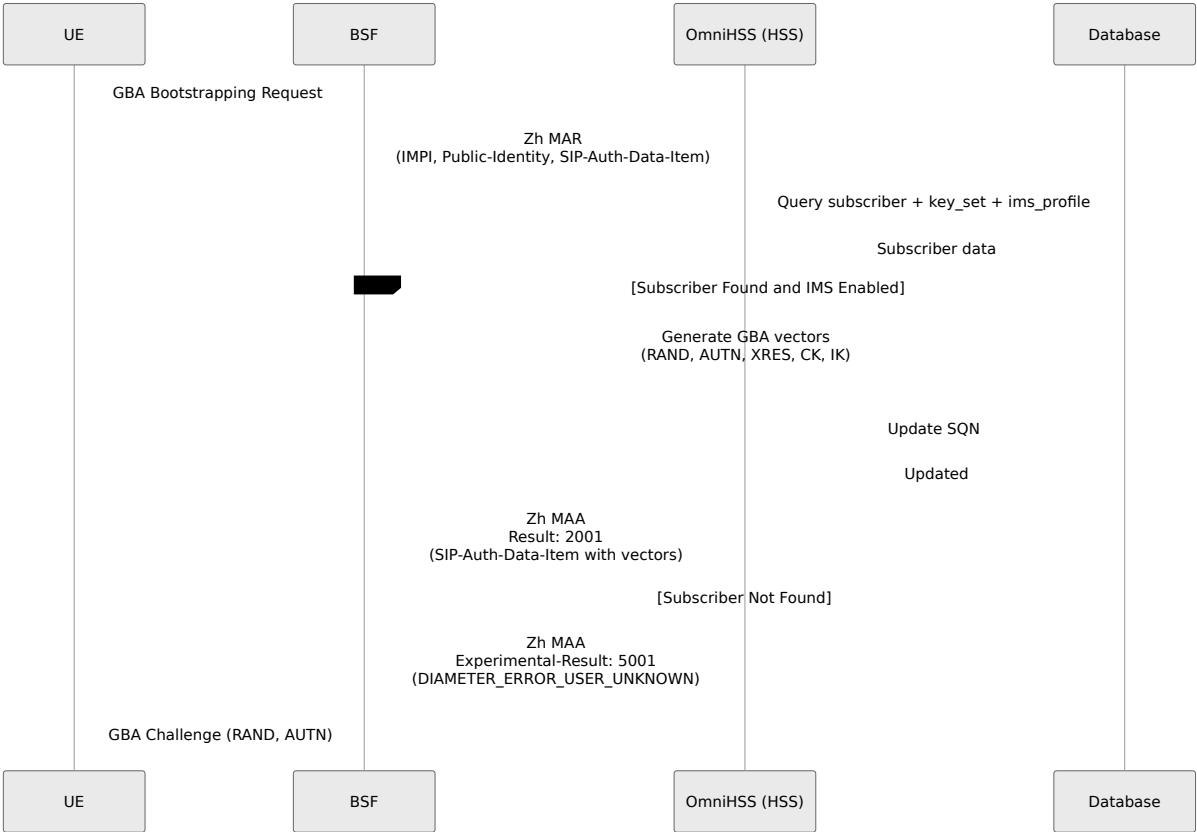


# Zh ( )

Zh (BSF) HSS (GBA)

# (MAR/MAA)

BSF GBA



AVP

- User-Name (IMPI)Public-IdentitySIP-Auth-Data-ItemDestination-Realm
- SIP-Auth-Data-ItemSIP-AuthenticateSIP-AuthorizationConfidentiality-KeyIntegrity-KeyGBA-UserSecSettings

GBA

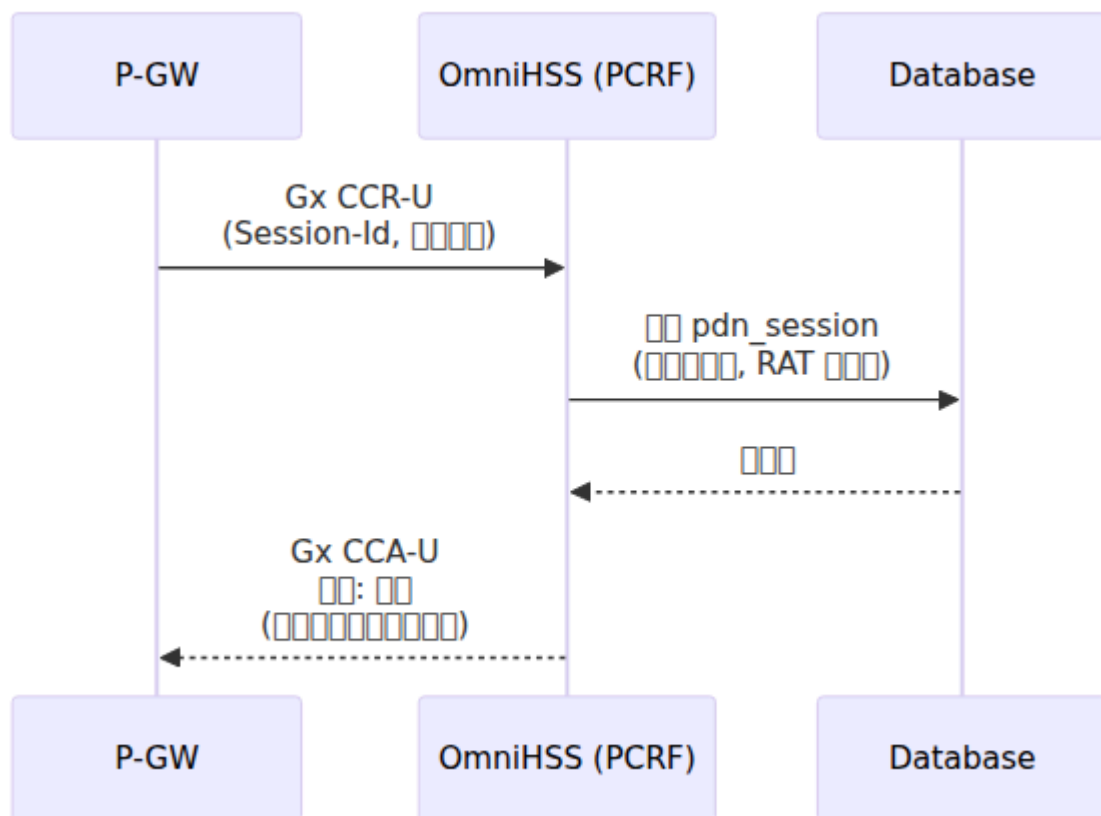
| MBMS     |     |
|----------|-----|
| HTTPS    | GBA |
| NAF      |     |
| Presence |     |

# SWm ( ( 3GPP )

SWm 3GPP AAA HSS 3GPP ePDG WiFi  
EAP-AKA

## Diameter-EAP- (DER/DEA)

WiFi EAP

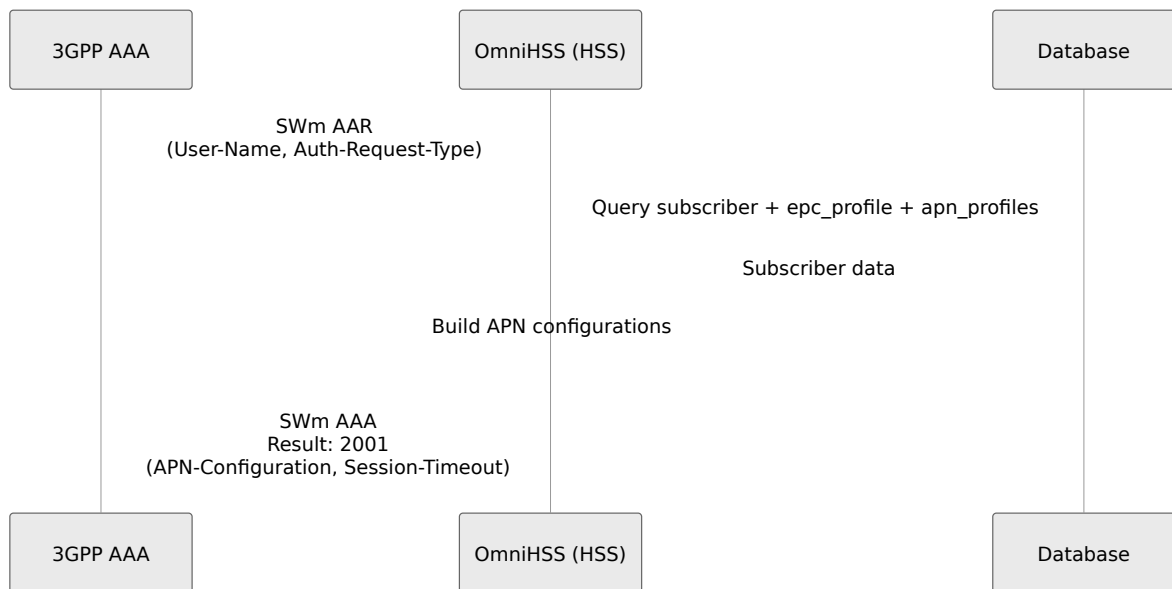


AVP

- Session-IdAuth-Application-IdEAP-PayloadAuth-Request-TypeUser-Name
- EAP-PayloadEAP-Master-Session-Key (MSK)APN-ConfigurationSession-Timeout

## AA- (AAR/AAA)

EAP

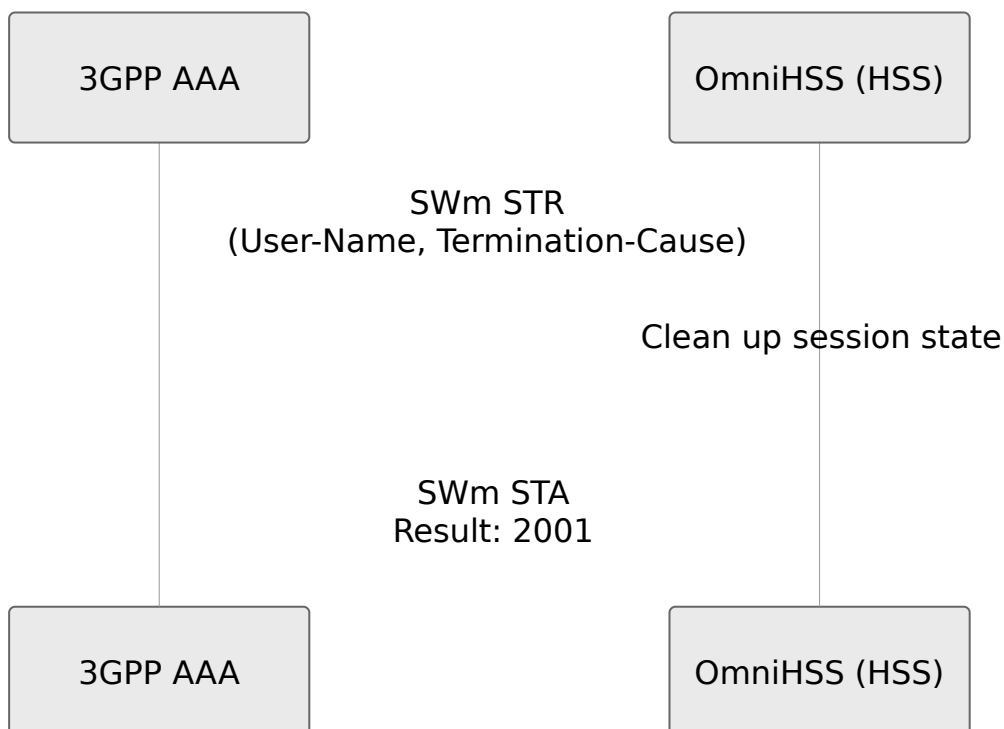


## AVP

- Session-IdAuth-Application-IdUser-NameAuth-Request-TypeAAR-Flags
- APN-Configuration3GPP-Charging-CharacteristicsSession-Timeout

## STR/STA

WiFi



メッセージ

| メッセージ                         | 番号 | 説明            |
|-------------------------------|----|---------------|
| DIAMETER_LOGOUT               | 1  | ログアウト         |
| DIAMETER_SERVICE_NOT_PROVIDED | 2  | サービスが提供されていない |
| DIAMETER_BAD_ANSWER           | 3  | 不正な応答         |
| DIAMETER_ADMINISTRATIVE       | 4  | 管理メッセージ       |
| DIAMETER_LINK_BROKEN          | 5  | リンクが切れた       |
| DIAMETER_AUTH_EXPIRED         | 6  | 認証が有効期限切れ     |
| DIAMETER_USER_MOVED           | 7  | ユーザーが移動した     |
| DIAMETER_SESSION_TIMEOUT      | 8  | セッションがタイムアウト  |

メッセージの送信

メッセージ (S6a AIR)

送信

- 送信元 (Ki, OPC, AMF)
- SQN 送信元
- 送信先

メッセージ (S6a ULR)

送信

- EPC 送信元 APN



2. 電話番号
3. MME 電話番号

## IMS 番号 (Cx SAR)

番号

1. IMS 電話番号
2. IFC 番号 XML
3. S-CSCF 番号
4. 電話番号 MSISDN

## PDN 番号 (Gx CCR-I)

番号

1. APN 番号 EPC 番号 APN 番号
2. APN QoS 番号
3. PDN 番号

---

← 番号

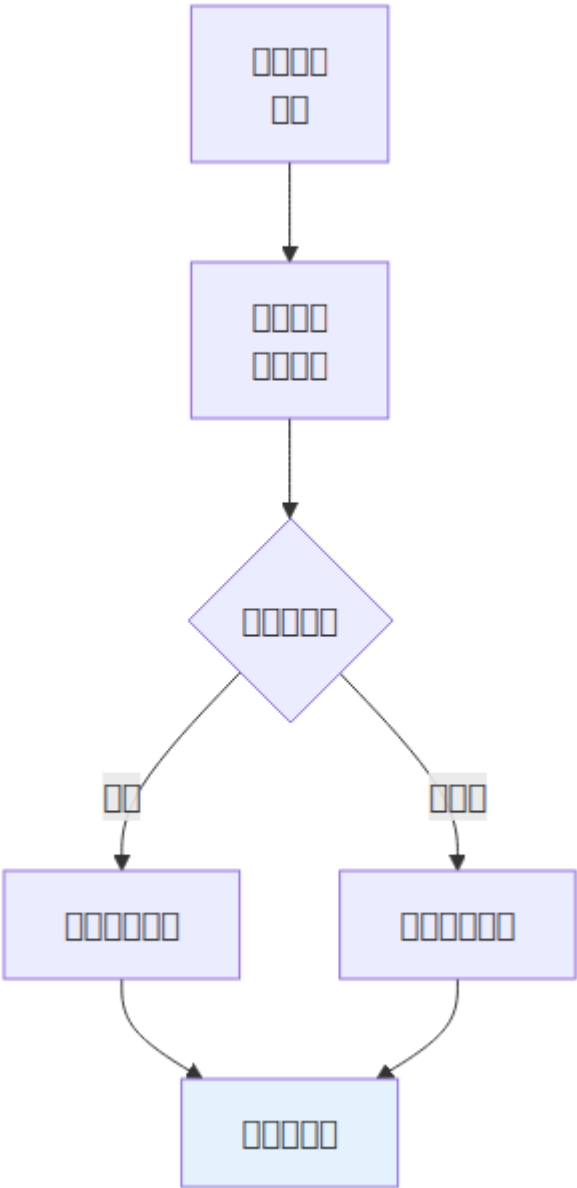
# OmniHSS 架构图

← 架构图

图

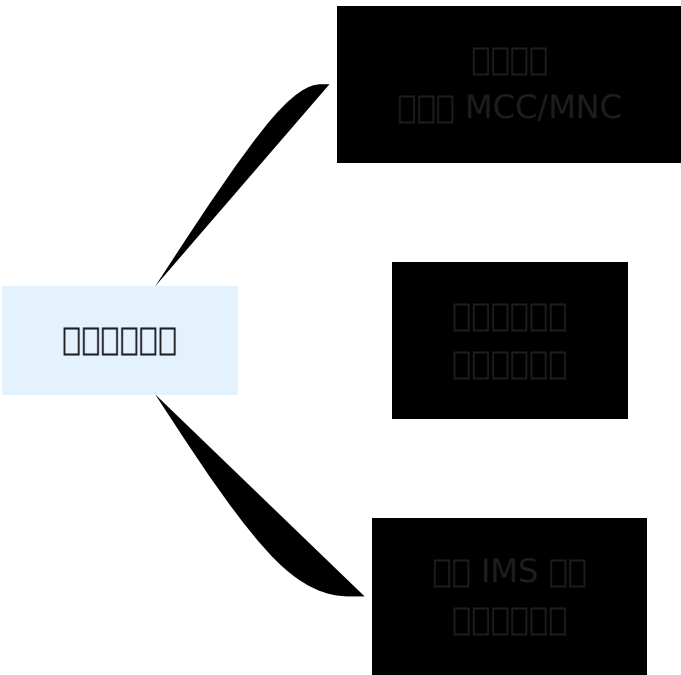
OmniHSS 架构图展示了 IMS 网络中的核心组件及其交互流程。

架构图



# IMS 网络

网络



## 网络

网络MNC/MCC 网络

网络:

- name - 网络
- mcc - 网络3 网络
- mnc - 网络2-3 网络
- data\_action - "allow" 或 "deny"
- ims\_action - "allow" 或 "deny"

## 网络

网络

例:

- `name` - 名前
  - `data_action_if_no_rules_match` - "allow" 或 "deny"
  - `ims_action_if_no_rules_match` - "allow" 或 "deny"
- 

例

例

```
例
curl -k -X POST https://hss.example.com:8443/api/roaming/profile \
-H "Content-Type: application/json" \
-d '{
 "roaming_profile": {
 "name": "例",
 "data_action_if_no_rules_match": "allow",
 "ims_action_if_no_rules_match": "allow",
 "roaming_rules": []
 }
}'
```

例

```
例
curl -k -X POST https://hss.example.com:8443/api/roaming/profile \
-H "Content-Type: application/json" \
-d '{
 "roaming_profile": {
 "name": "例",
 "data_action_if_no_rules_match": "deny",
 "ims_action_if_no_rules_match": "deny",
 "roaming_rules": []
 }
}'
```

## API Examples

# AT&T

```
RULE1=$(curl -k -X POST
https://hss.example.com:8443/api/roaming/rule \
-H "Content-Type: application/json" \
-d '{
 "roaming_rule": {
 "name": "AT&T",
 "mcc": "310",
 "mnc": "410",
 "data_action": "allow",
 "ims_action": "allow"
 }
}' | jq -r '.response.id')
```

# Verizon

```
RULE2=$(curl -k -X POST
https://hss.example.com:8443/api/roaming/rule \
-H "Content-Type: application/json" \
-d '{
 "roaming_rule": {
 "name": "Verizon",
 "mcc": "311",
 "mnc": "480",
 "data_action": "allow",
 "ims_action": "allow"
 }
}' | jq -r '.response.id')
```

#

```
curl -k -X POST https://hss.example.com:8443/api/roaming/profile \
-H "Content-Type: application/json" \
-d "{
 \"roaming_profile\": {
 \"name\": \"\",
 \"data_action_if_no_rules_match\": \"deny\",
 \"ims_action_if_no_rules_match\": \"deny\",
 \"roaming_rules\": [$RULE1, $RULE2]
 }
}"
```

□□□□□□□□

```
□□□□□□□□ IMS □□□
curl -k -X POST https://hss.example.com:8443/api/roaming/rule \
-H "Content-Type: application/json" \
-d '{
 "roaming_rule": {
 "name": "□□□□ - T-Mobile",
 "mcc": "310",
 "mnc": "260",
 "data_action": "allow",
 "ims_action": "deny"
 }
}'
```

## API 调用

```
创建漫游规则
RULE=$(curl -k -X POST
https://hss.example.com:8443/api/roaming/rule \
-H "Content-Type: application/json" \
-d '{
 "roaming_rule": {
 "name": "漫游规则",
 "mcc": "206",
 "mnc": "01",
 "data_action": "deny",
 "ims_action": "deny"
 }
}' | jq -r '.response.id')

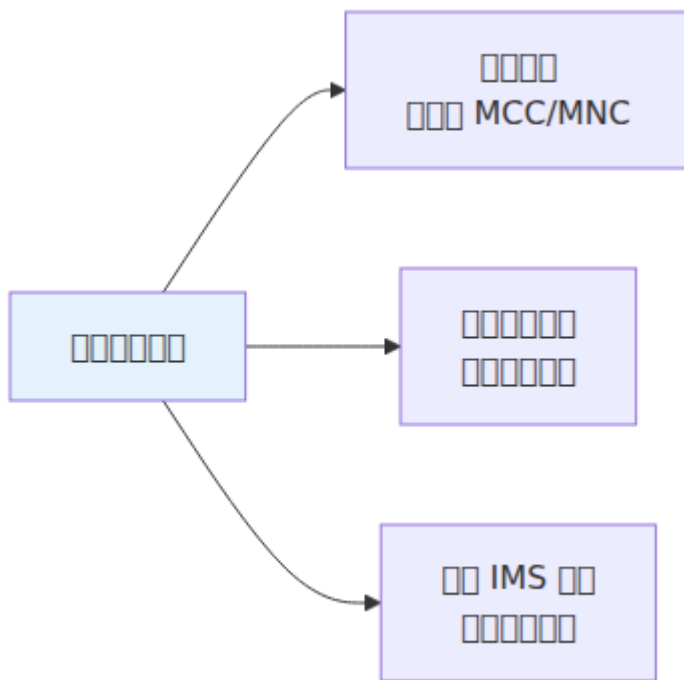
创建漫游配置文件
curl -k -X POST https://hss.example.com:8443/api/roaming/profile \
-H "Content-Type: application/json" \
-d "{
 \"roaming_profile\": {
 \"name\": \"漫游配置文件\",
 \"data_action_if_no_rules_match\": \"allow\",
 \"ims_action_if_no_rules_match\": \"allow\",
 \"roaming_rules\": [$RULE]
 }
}"
```

---

## 配置

### 配置 1

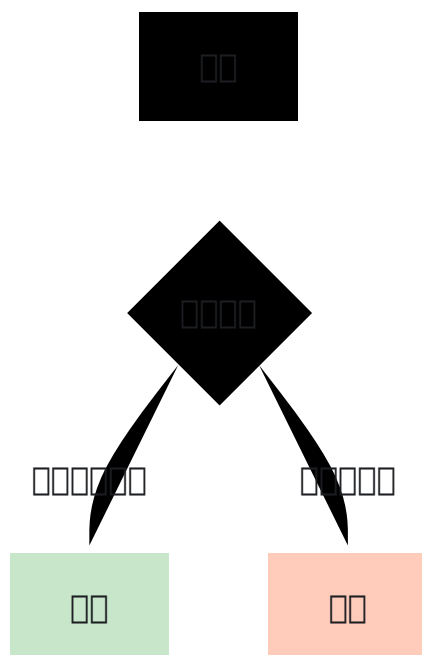
配置 1 的详细信息



：

- 
- MCC 310, 311, 312, 313, 314, 315, 316

2



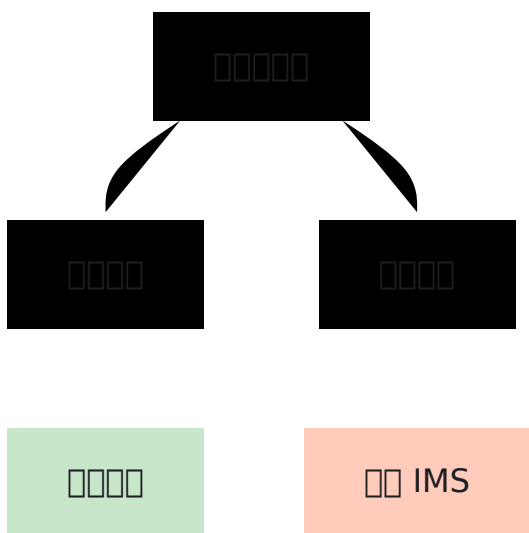
：



- 网络类型
- 网络类型标识符 MCC/MNC

### 第 3 步

网络类型标识符 MCC/MNC



网络类型:

- 网络类型 `data_action: "allow"` `ims_action: "deny"`

### 第 4 步

网络类型标识符

网络类型标识符 MME/网络类型标识符OmniHSS 网络类型标识符

# MCC/MNC

## 미국 MNC

| MCC     | 국가   | 통신사                        |
|---------|------|----------------------------|
| 310-316 | 미국   | AT&T, Verizon, T-Mobile    |
| 302     | 캐나다  | Rogers, Bell, Telus        |
| 234-235 | 영국   | Vodafone, O2, EE           |
| 262     | 독일   | Deutsche Telekom, Vodafone |
| 208     | 프랑스  | Orange, SFR, Bouygues      |
| 222     | 이탈리아 | TIM, Vodafone, Wind        |
| 214     | 스페인  | Movistar, Vodafone         |

## 미국 MNC 310-316

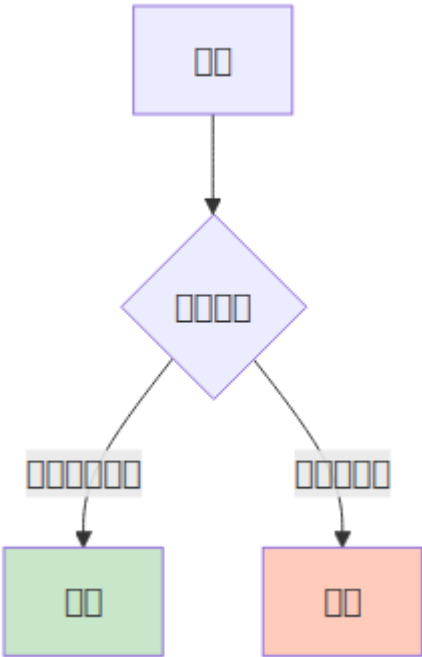
| MCC | MNC | 통신사      |
|-----|-----|----------|
| 310 | 410 | AT&T     |
| 311 | 480 | Verizon  |
| 310 | 260 | T-Mobile |
| 310 | 120 | Sprint   |
| 313 | 380 | 미국군      |

참고: ITU-T E.212 - MCC/MNC

□□□□□

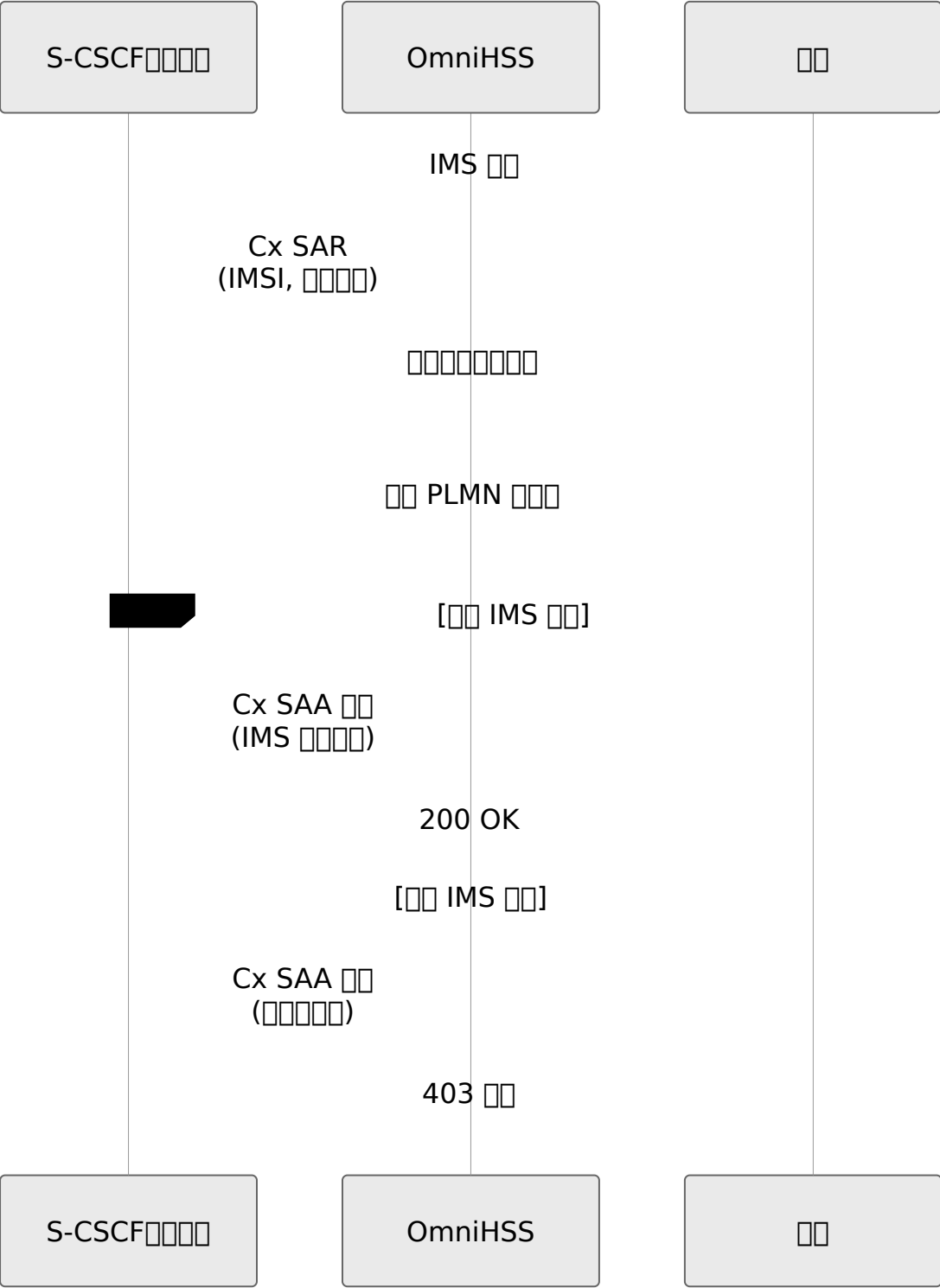
**S6a** □□□□□

□□□□□□□□□□



**Cx** □□□IMS□

□□□□□□□□□ IMS □□



00000000

000000000000

00000000:

- 0000000000000000
- 00000000000000

0000000000:

- 000000000000 MCC/MNC 000000
- 0000000000000000
- 00000000 data\_action 0

000000000000 **IMS**

0000 **IMS** 00:

- 0000000000
- 00 data\_action 0 ims\_action 0
- 00000000 IMS 000000

00000000

00000000:

[info] 0000: IMSI 001001123456789, 00 PLMN 310-410  
[info] 00000000: "00 AT&T"  
[info] 0000: 00, IMS 00: 00

## □□□□

### □□□□□□

1. □□□□ - □□□□□□□□□□□□
2. □□□□ - □□□□□□□□□□□□
3. □□□□ - □□□□□□□□□□□□
4. □□□□ - □□□□□□□□□□□□

### □□□□

1. □□□□□□ - "Allow-ATT-Data-Only" □□□ "Rule1"
2. □□ **MCC/MNC** - □□□□□□□□□□□□
3. □□□□□□ - □□□□□□□ IMS
4. □□□□□□ - □□□□□□□□□□□□

### □□□□

1. □□□□ - □□□□□□/□□□□□□□□
2. □□□□ - □□□□□□□□□□□□□□□□
3. □□ - □□□□□□□□□□□□□□
4. □□□□ - □□□□□□□□□□□□

# Sh RepositoryData ( )

OmniHSS (AS) Sh MMTEL-  
Services/OIP/OIR IMS-ODB-Information  
AS HSS

“ ” 3GPP TS 29.328 HSS AS  
XMLOmniHSS ServiceData AS  
etags XCAP/MMTel

REST Provisioning API /CRM

## 

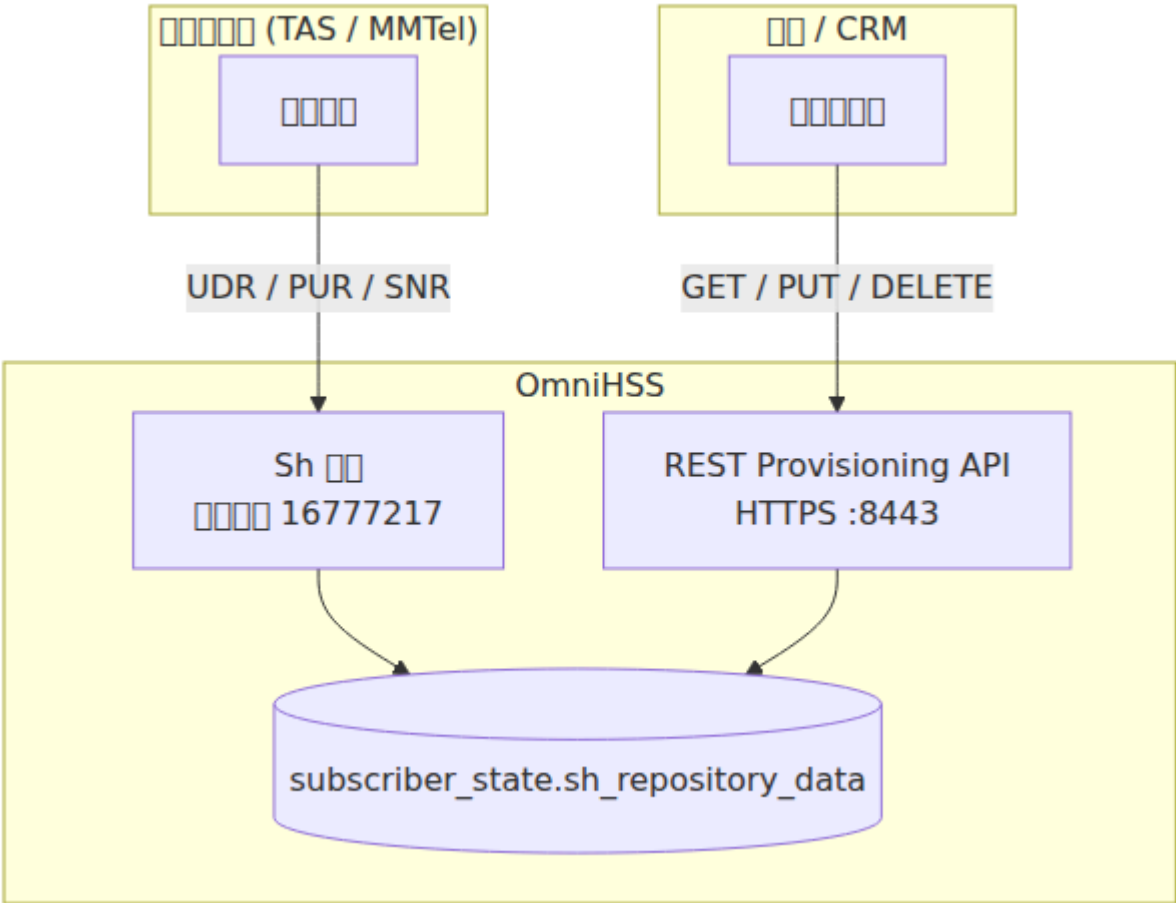
- 
- 
- Sh
- SequenceNumber
- REST Provisioning API
- 
- 

## 

Service-Indication“ ”

| 項目                 | 項目名    | 説明                                 |
|--------------------|--------|------------------------------------|
| Service-Indication | HSS 項目 | MMTEL-Services IMS-ODB-Information |
| SequenceNumber     | HSS 項目 | SequenceNumber                     |
| ServiceData        | AS 項目  | XML                                |

HSS 項目 Service-Indication SequenceNumber ServiceData



MMTEL

subscriber\_state.sh\_repository\_data Service-Indication JSON



```
{
 "MMTEL-Services": { "sequence_number": "0", "service_data":
"<MMTelServices ...>...</MMTelServices>" },
 "IMS-ODB-Information": { "sequence_number": "1", "service_data":
"<OdbForImsOrientedServices xmlns=\"odb.mmtel.omnitou.ch\"/>" }
}
```

| 項目              | 説明                                                         |
|-----------------|------------------------------------------------------------|
| sequence_number | シーケンス番号 HSS から UDA に提供される                                  |
| service_data    | XML 形式で AS から提供される Service-Indication<br>ServiceData 要素を含む |

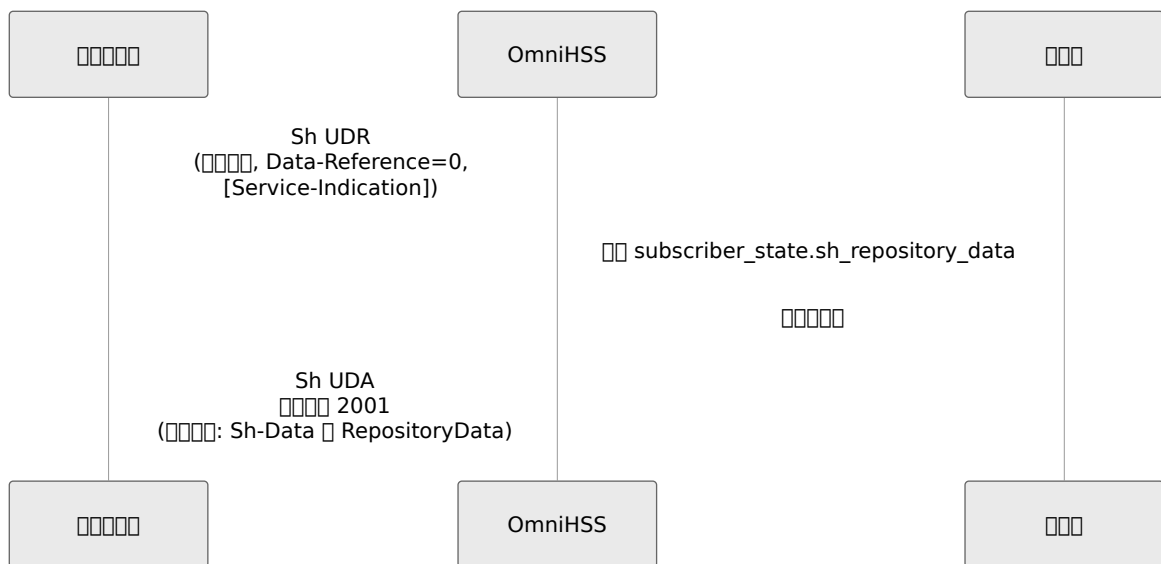
シーケンス番号 HSS から提供される

## Sh 処理

RepositoryData 要素 Sh-Data XML 要素 User-Data AVP 要素 **Data-Reference**  
0 要素 RepositoryData 要素 **Service-Indication** AVP 要素

### シーケンス (UDR/UDA) 処理

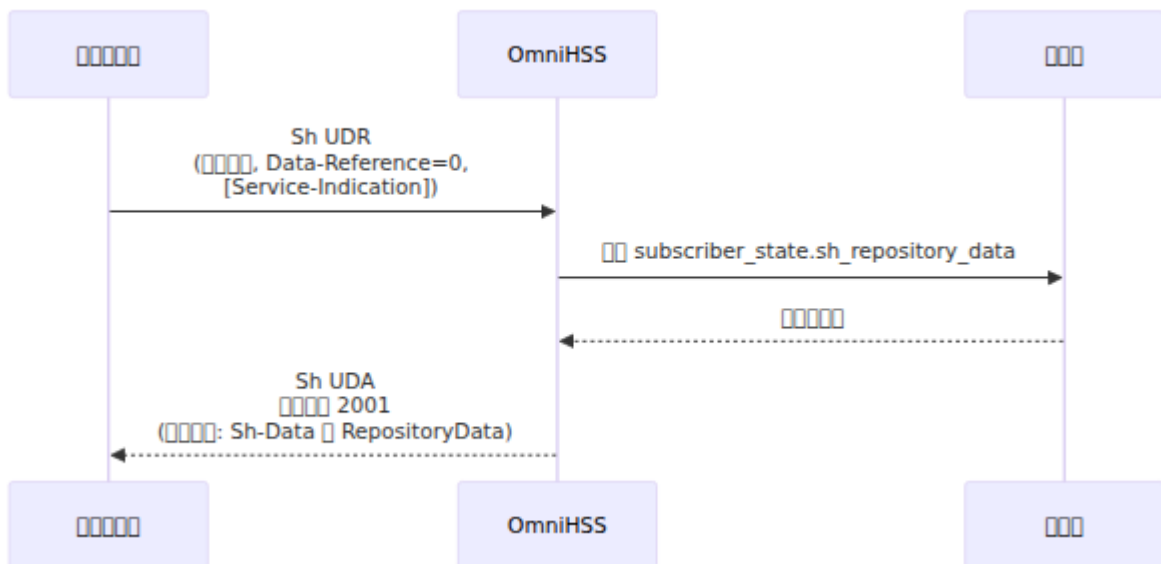
AS 要素 Service-Indication 要素  
<RepositoryData> 要素



IMS Experimental-Result 5001  
(DIAMETER\_ERROR\_USER\_UNKNOWN)

(PUR/PUA)

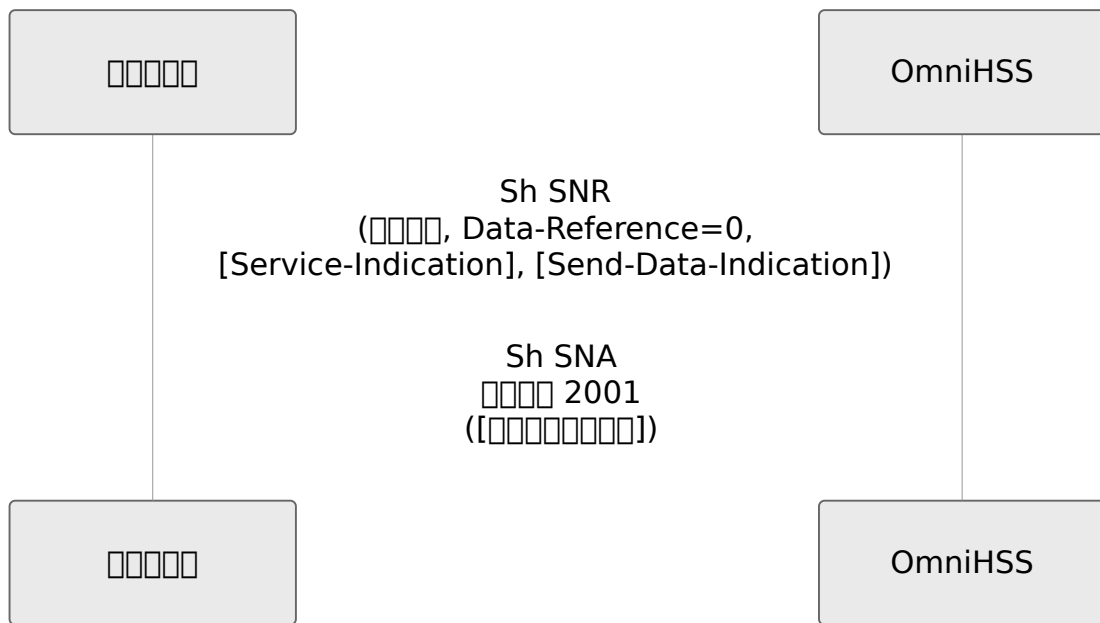
AS ServiceData SequenceNumber



PUR

(SNR/SNA)

AS Send-Data-Indication OmniHSS SNR RepositoryData



## SequenceNumber

SequenceNumber 3GPP TS 29.328

- SequenceNumber 0 0
- SequenceNumber + 1
- 65535 1 0 0
- SequenceNumber Experimental-Result 5105 (DIAMETER\_ERROR\_TRANSPARENT\_DATA\_OUT\_OF\_SYNC)



Absent

PUR SequenceNumber  
0 (00)

Stored0

PUR SequenceNumber  
1 (00+1)      DELETE (REST)

Stored1

PUR SequenceNumber  
n+1

StoredN

PUR 000  
SequenceNumber  
→ 5105, 000

000 0000000 SequenceNumber 0 000 SequenceNumber 5 0 PUR 000000 5105 0  
0000000000 SequenceNumber 1 0 PUR 0000000000 1 0

## REST Provisioning API

000000000000/CRM 000000000000 IMSI 000000000000 service\_indication 00  
000000000000/0000000000000000 service\_data 00000000 Sh 000

00000000 API 000

```
{ "status": "success", "response": { } }
```

Sh REST API SequenceNumber

GET /api/subscriber/repository\_data/:imsi

```
curl -k
https://hss.example.com:8443/api/subscriber/repository_data/001001123
```

```
{
 "status": "success",
 "response": {
 "MMTEL-Services": { "sequence_number": "0",
 "service_data": "<MMTelServices .../>" },
 "IMS-ODB-Information": { "sequence_number": "1",
 "service_data": "<OdbForImsOrientedServices
xmlns=\"odb.mmtel.omnitou.ch\"/>" }
 }
}
```

Service-Indication

GET /api/subscriber/repository\_data/:imsi?  
service\_indication=:service\_indication

```
curl -k
"https://hss.example.com:8443/api/subscriber/repository_data/001001123
service_indication=MMTEL-Services"
```

```
{
 "status": "success",
 "response": { "sequence_number": "0", "service_data": "<MMTelServices .../>" }
}
```

□□□□□□□□

□□□□□□□□□□□□□□□□

□□□ PUT /api/subscriber/repository\_data/:imsi

```
curl -k -X PUT
https://hss.example.com:8443/api/subscriber/repository_data/001001123
\
-H "Content-Type: application/json" \
-d '{
 "service_indication": "MMTEL-Services",
 "service_data": "<MMTelServices version=\"1\"/>",
 "sequence_number": "0"
}'
```

| □□                 | □<br>□      | □<br>□ | □<br>□ | □□                                                                            |
|--------------------|-------------|--------|--------|-------------------------------------------------------------------------------|
| service_indication | □<br>□<br>□ | □      | -      | □□□/□□□□□□□□□□ MMTEL-Services □□                                              |
| service_data       | □<br>□<br>□ | □      | -      | □□□□ XML □□□□□□□□□□□□□□□□□□□□□□□□<br>Service-Indication □<br>SequenceNumber□□ |
| sequence_number    | □<br>□<br>□ | □      | 0      | □□□□□□□□□□                                                                    |

成功レスポンス

```
{
 "status": "success",
 "response": { "sequence_number": "0", "service_data": "<MMTelServices version=\"1\"/>" }
}
```

削除

Service-Indication 削除

DELETE /api/subscriber/repository\_data/:imsi

```
curl -k -X DELETE
https://hss.example.com:8443/api/subscriber/repository_data/001001123
\
-H "Content-Type: application/json" \
-d '{ "service_indication": "MMTEL-Services" }'
```

| 項目                 | 値     | 単位 | 範囲 | 備考             |
|--------------------|-------|----|----|----------------|
| service_indication | MMTEL | 文字 | -  | MMTEL-Services |

削除成功

参照

Data-Reference (RepositoryData)

| ID | 参照先            | 参照先名                  | 参照先バージョン                |
|----|----------------|-----------------------|-------------------------|
| 0  | RepositoryData | Service-Indication 削除 | 3GPP TS 29.328<br>\$7.6 |

❓❓❓ 表

| ID   | Message Name                                | Notes                          |
|------|---------------------------------------------|--------------------------------|
| 2001 | DIAMETER_SUCCESS                            | UDR/PUR/SM<br>OK               |
| 5001 | DIAMETER_ERROR_USER_UNKNOWN                 | Unknown user                   |
| 5105 | DIAMETER_ERROR_TRANSPARENT_DATA_OUT_OF_SYNC | Sequence Number<br>out of sync |

表 ID

| ID       | Sh | AS ↔ HSS 表 | 3GPP TS 29.329 |
|----------|----|------------|----------------|
| 16777217 | Sh | AS ↔ HSS 表 | 3GPP TS 29.329 |

表 Service-Indications

| Service-Indication  | Table Name                                  |
|---------------------|---------------------------------------------|
| MMTEL-Services      | MMTel 表 / SIP / OIP / OIR 表<br>simservs XML |
| IMS-ODB-Information | IMS 表                                       |

Service-Indication 表 XML 表 HSS 表



|  |
|--|
|  |
|--|

□□□□□□□□□□ **5105**

000 PUR 0000 Experimental-Result 5105 00000000

□□□□□

- AS SequenceNumber HSS
- AS SequenceNumber

□ □ □ □ □

1. `UDR` `SequenceNumber`
2. `PUR` `SequenceNumber` = `UDR` + 1
3. `REST API` `API`

## ServiceData

AS HSS

5/5

- OmniHSS ServiceData AS XML

□ □ □ □ □

1. UDA/REST
2. AS XML

REST

```
GET .../repository_data/:imsi?service_indication=...
```

□□□□□

- Service-Indication

- IMSI 0000000000

000000

1. GET /api/subscriber/repository\_data/:imsi 0000000000000000
2. 00 IMSI 0000 00 IMSI

# SRVCC - 3GPP TS 23.216

← 3GPP TS 23.216

SRVCC

SRVCC 3GPP TS 23.216 LTE VoLTE 2G/3G CS MME Sv TS 29.280 Sv PS CS MME HSS

| AVP      | Code | Description                                          |
|----------|------|------------------------------------------------------|
| STN-SR   | 1433 | SRVCC MSC-Server E.164 TAS/S-CSCF SIP REFER VoLTE CS |
| C-MSISDN | 1452 | MSISDN TAS STN-SR                                    |

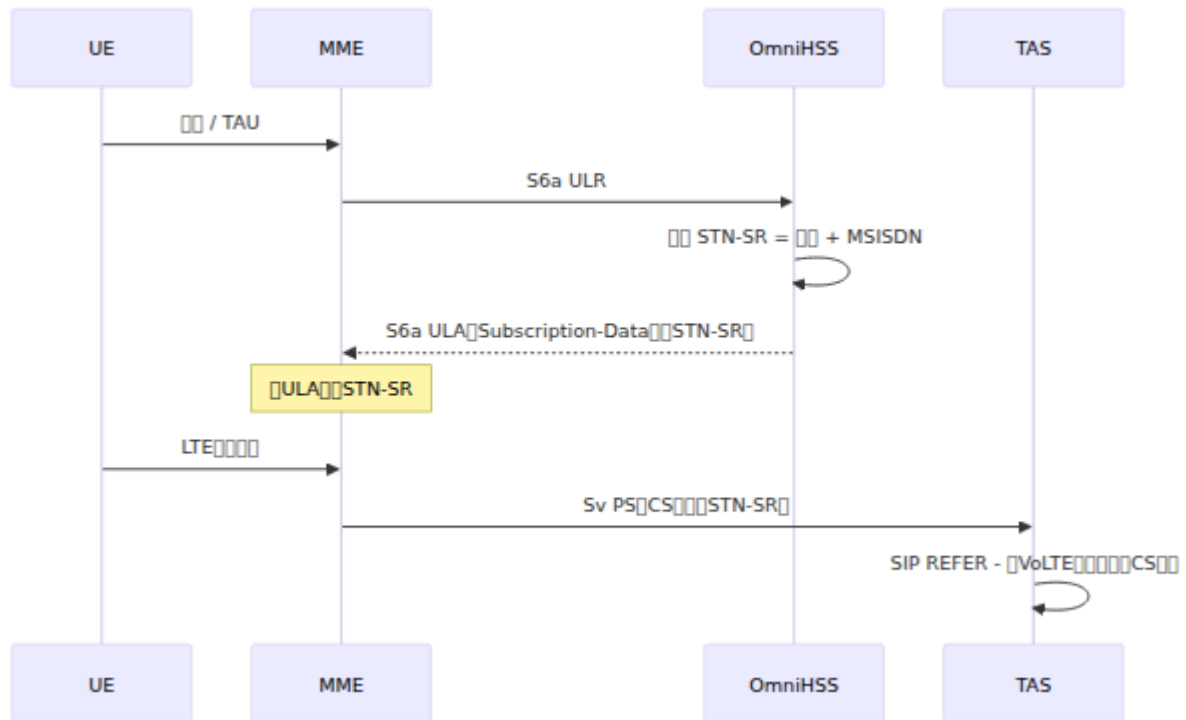
OmniHSS S6a ULA Subscription-Data AVP STN-SR MME ULA Sv PS CS

Table

| Code      | Description                                 |
|-----------|---------------------------------------------|
| TS 23.216 | SRVCC PS CS                                 |
| TS 29.272 | S6a STN-SR §7.3.89 Subscription-Data §7.3.2 |
| TS 23.003 | STN-SR §13.4                                |

## Sequence

OmniHSS provides STN-SR to HSS ULA and MSISDN. STN-SR is TBCD Subscription-Data STN-SR AVP.



OmniHSS provides HSS STN-SR to SRVCC.

OmniHSS provides STN-SR AVP ULA to SRVCC.

## Config

srvcc config/runtime.exs :hss

```
config :hss, :srvcc,
 stn_sr_prefix: "9"
```

00

| 00            | 00 | 00 | 00   | 00                                                                                                  |
|---------------|----|----|------|-----------------------------------------------------------------------------------------------------|
| stn_sr_prefix | 00 | 00 | 0000 | 00000000STN-SR0000000000000000MSISDN000000E.164 STN-SR000000000000OmniHSS000000ULA000000STN-SR AVP0 |

STN-SR0000

000STN-SR0000

STN-SR = stn\_sr\_prefix + 000 MSISDN

0000**TBCD**0000000000000000**TS 24.008 §10.5.4.7**0000000000STN-SR AVP00014330000104150000

000

| 00         | 0            |
|------------|--------------|
| 000000     | 9            |
| 000 MSISDN | 61412345678  |
| 000 STN-SR | 961412345678 |

MSC-Server00 961412345678 00000TAS0TAS00MSISDN00000000IMS0000000SIP REFER000000CS0000

00000

00**SRVCC**

```
config/runtime.exs
```

```
config :hss, :srvcc,
 # MSC-Server STN-SR = stn_sr_prefix + MSISDN
 # STN-SR = stn_sr_prefix + MSISDN
 stn_sr_prefix: "9"
```

ULA Subscription-Data STN-SR MSISDN TBCD  
MME SRVCC Sv PS CS

SRVCC

srvcc stn\_sr\_prefix OmniHSS STN-SR AVP ULA  
SRVCC

## MME

MME OmniMME ULA Subscription-Data STN-SR  
SRVCC MSC-Server Sv PS CS STN-SR MME  
Omnitouch/omnimme#426

OmniHSS  
stn\_sr\_prefix = 9  
56a ULAinSubscription-Data  
STN-SR = 961412345678

OmniMME  
STN-SR  
Sv PS[CS] STN-SR = 961412345678

MSC-Server  
961412345678

TAS / S-CSCF  
SIP REFERIN(VoLTE)  
CS

## ULA STN-SR

MME Sv PS CS SRVCC

- config/runtime.exs stn\_sr\_prefix

- HSS

1. `config/runtime.exs` `srvcc` `stn_sr_prefix`
2. OmniHSS
3. MME ULR TAU ULA `Subscription-Data` STN-SR

## STN-SR

MSC-Server

- 
- HSS MSISDN E.164 +

1. API MSISDN `GET /api/subscriber/imsi/:imsi`
2. STN-SR `stn_sr_prefix` MSISDN
3. MSC-Server

---

←

# Sy

Sy HSS **PCRF** **OCS** PCRF  
OCS

OmniHSS PCRF **Gx** QoS **Rx** IMS Sy OCS  
Gx

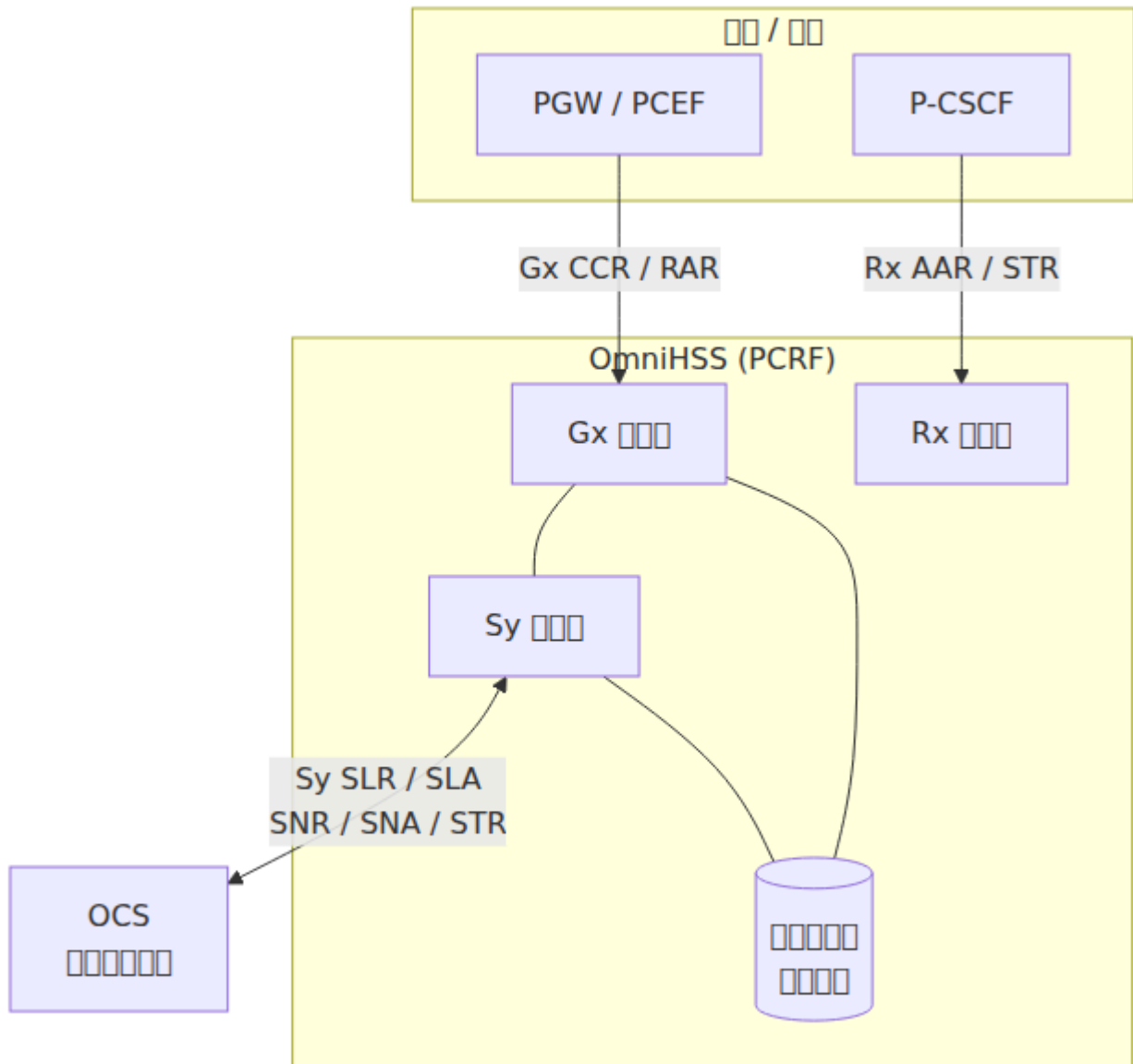
Sy **3GPP TS 29.219** ID **16777302**

| ID  | 16777302              |
|-----|-----------------------|
|     | <b>3GPP TS 29.219</b> |
| HSS | PCRF SLR/STR SNR      |
|     | OCS                   |
|     | TCP SCTP 3868         |

Sy Gy Sy Gy (**TS 32.299**) PGW OCS Sy  
**PCRF** OCS



11



HSS Sy Sy OCS Sy  
PCRFGx

□ □ □ □ □ □ □ □ □ □

OCS    □□□□ □□□□□□ □□□□□□□□□□□□

- 000000000000AVP 2901 - 000000000000
- 000000000000AVP 2902 - 0000000000000000 active over\_quota throttled  
000000000000 3GPP 000000000000 OCS PCRF 00000000TS 29.219 \$3.10

OmniHSS 数据库表结构

| Sy AVP                           | 数据类型              | 描述           | 备注       |
|----------------------------------|-------------------|--------------|----------|
| Policy-Counter-Identifier (2901) | INTEGER (1-65535) | HSS 策略计数器标识符 | 策略计数器标识符 |
| Policy-Counter-Status (2902)     | INTEGER (1-255)   | 策略计数器状态      | 策略计数器状态  |

策略计数器标识符和策略计数器状态在“策略计数器”表中维护。

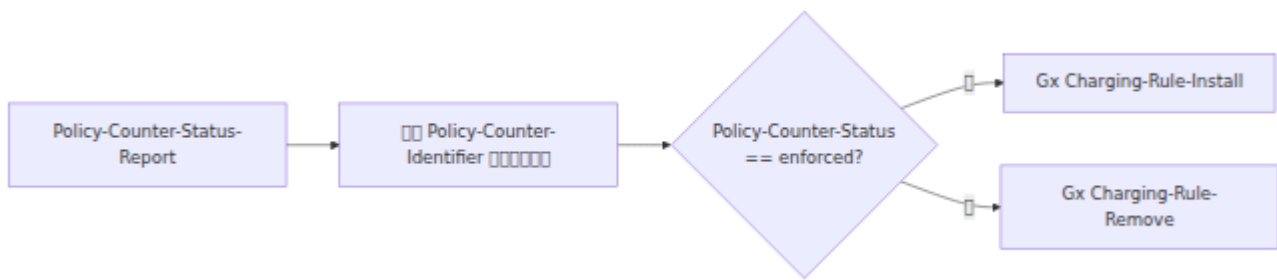
- 策略计数器标识符 **enforced** 策略计数器标识符 Gx 策略计数器 **Charging-Rule-Install**
- 策略计数器标识符 **Charging-Rule-Remove**

**Sy** 策略计数器标识符和策略计数器状态 Sy 策略计数器标识符 QoS 策略 AVP - 策略计数器 Gx 策略计数器 OCS 策略 PCRF 策略计数器标识符 策略计数器标识符 策略计数器标识符 HSS 策略计数器标识符 策略计数器标识符 **Policy-Counter-Identifier** 策略计数器标识符 OCS 策略计数器标识符 **enforced** 策略计数器标识符



## Sy 策略计数器

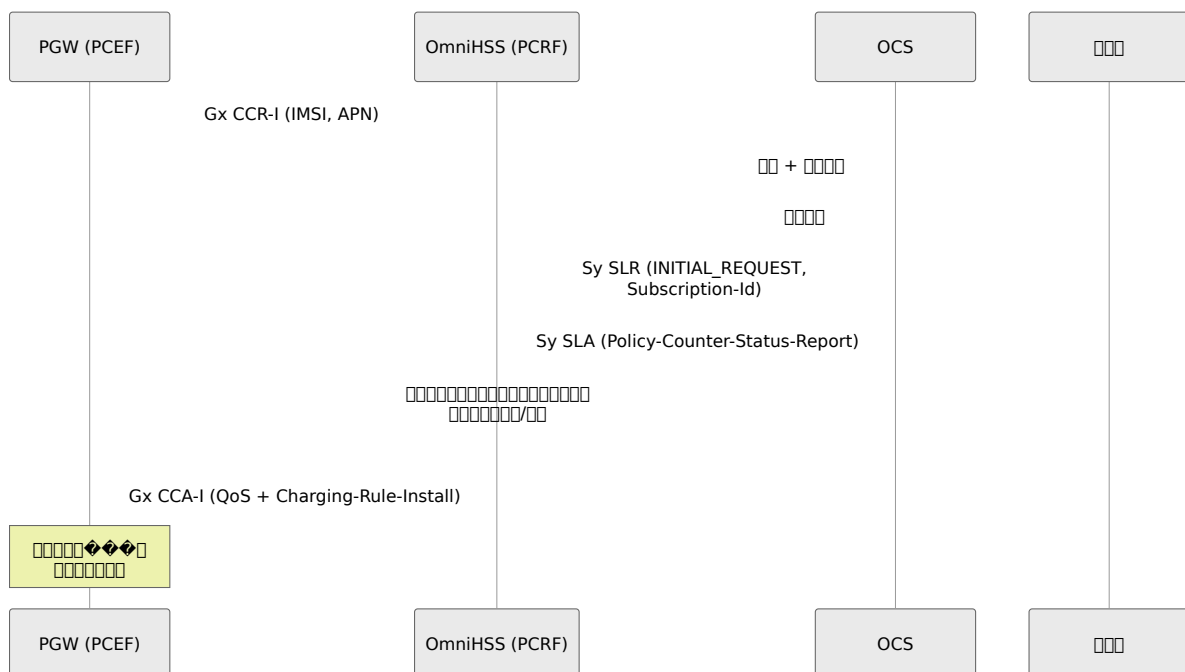
Sy 策略计数器标识符和策略计数器状态 PCRF 策略计数器标识符 Sy 策略计数器标识符 **Session-Id** 策略计数器标识符 TS 29.219 §4.4.1



- **(SLR, INITIAL\_REQUEST)** PCRF → Gx CCR-I → OCS → SLA → Sy
- **(SLR, INTERMEDIATE\_REQUEST)** PCRF → Sy
- **(SNR / SNA)** OCS → PCRF → SNA → Gx
- **(STR / STA)** PCRF → Sy

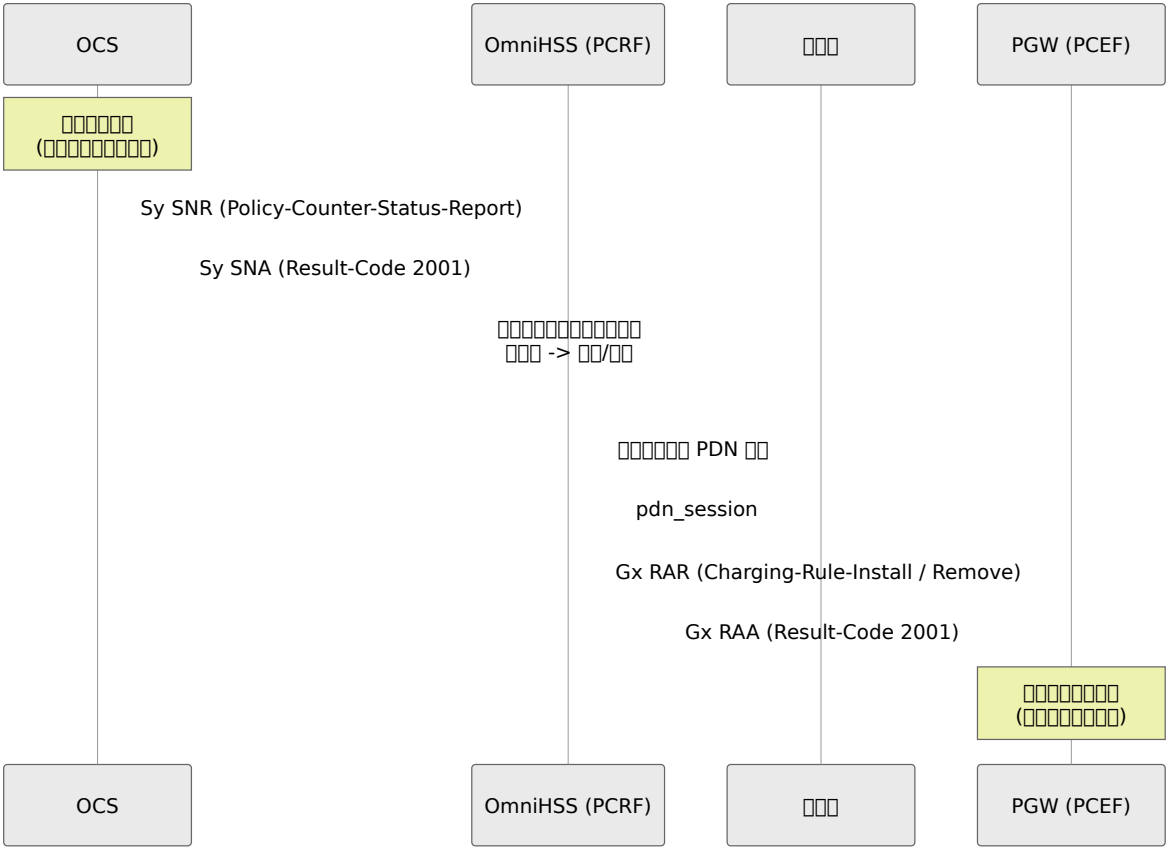
1.1

## 1. PDN



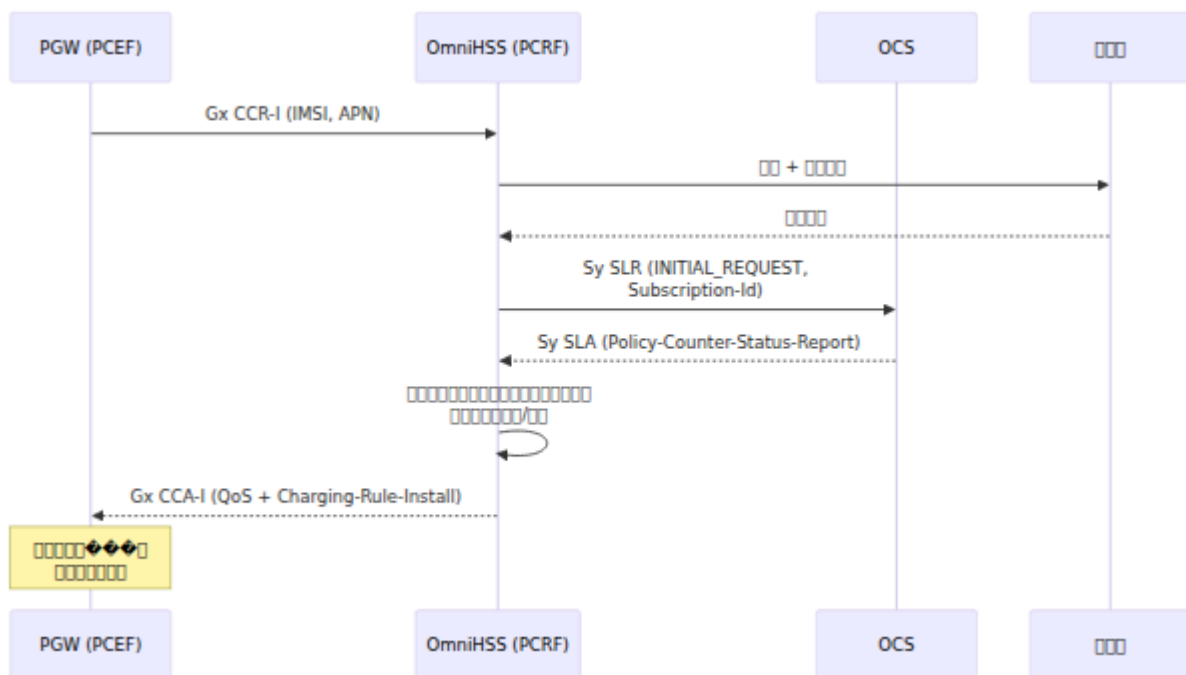
PCRF → Gx CCA-I → OCS → PCRF → APN → Sy

**2 OCS -> PCRF -> Gx RAR**



3GPP PCRF 3GPP Gx 3GPP 3GPP VoLTE 3GPP 3GPP PGW 3GPP  
 3GPP CCR

## 3 Sy



## Sy

Sy HSS Gx Rx config/runtime.exs

```
%{
 application_name: :sy,
 application_dictionary: :diameter_gen_3gpp_sy,
 vendor_specific_application_ids: [
 %{vendor_id: 10415, auth_application_id: 16_777_302,
 acct_application_id: nil}
]
}
```

| Field                           | Type             | Required | Default | Description                          |
|---------------------------------|------------------|----------|---------|--------------------------------------|
| application_name                | Atom             | Yes      | -       | Application name (Sy)                |
| application_dictionary          | Atom             | Yes      | -       | Application dictionary (Sy)          |
| vendor_specific_application_ids | List             | Yes      | -       | Vendor specific application IDs (Sy) |
| vendor_id                       | Integer          | Yes      | -       | Vendor ID (3GPP)                     |
| auth_application_id             | Integer          | Yes      | -       | Authentication application ID (Sy)   |
| acct_application_id             | Integer<br>  nil | Yes      | nil     | Accounting application ID (Sy)       |

## Sy

Sy :hss, :sy config/runtime.exs

```
config :hss,
 sy: %{
 enabled: false
 }
```

| Field   | Type    | Required | Default | Description                                      |
|---------|---------|----------|---------|--------------------------------------------------|
| enabled | Boolean | Yes      | false   | Whether Sy is enabled (true) or disabled (false) |

Policy-Counter-Status enforced OCS

## OCS

HSS Destination-Realm Destination-Host Sy ID 16777302 Sy - DRA OCS DRA OCS

Sy DRA HSS DRA Sy

- OCS Policy-Counter-Identifier HSS name enforced
- HSS

Sy HSS Prometheus

hss\_sy\_messages\_total : : Sy

- command - slr sla snr sna str sta
- direction - incoming OCS outgoing OCS
- result - sent received success error

hss\_sy\_policy\_updates\_total : : Sy Gx

- action - install remove

- result - success or error or Gx RAR or PGW or

or

```
or OCS or Sy or
rate(hss_sy_messages_total{command="snr", direction="incoming"}
[5m])

or or or or or
rate(hss_sy_policy_updates_total{action="install",
result="success"}[5m])

Sy Gx RAR or
sum(rate(hss_sy_policy_updates_total{result="error"}[5m]))
/ sum(rate(hss_sy_policy_updates_total[5m]))
```

or

or

| or | or  | or      | or          | or       |
|----|-----|---------|-------------|----------|
| or | SLR | 8388635 | PCRF -> OCS | or       |
| or | SLA | 8388635 | OCS -> PCRF | or       |
| or | SNR | 8388636 | OCS -> PCRF | or       |
| or | SNA | 8388636 | PCRF -> OCS | or       |
| or | STR | 275     | PCRF -> OCS | or Sy or |
| or | STA | 275     | OCS -> PCRF | or       |

or ID 16777302 or STR/STA (RFC 6733 or)



## AVPs

| AVP                                | Code | Type       | Description                                                        |
|------------------------------------|------|------------|--------------------------------------------------------------------|
| Policy-Counter-Identifier          | 2901 | UTF8String | OmniHSS Policy Counter                                             |
| Policy-Counter-Status              | 2902 | UTF8String | Policy Counter Status                                              |
| Policy-Counter-Status-Report       | 2903 | Grouped    | Policy Counter Status Report                                       |
| SL-Request-Type                    | 2904 | Enumerated | SL Request Type<br>(0) INITIAL_REQUEST<br>(1) INTERMEDIATE_REQUEST |
| Pending-Policy-Counter-Information | 2905 | Grouped    | Pending Policy Counter Information                                 |
| Pending-Policy-Counter-Change-Time | 2906 | Time       | Pending Policy Counter Change Time                                 |
| SN-Request-Type                    | 2907 | Unsigned32 | SNR 0 = SNR 1 = OCS Sy                                             |
| Subscription-Id                    | 443  | Grouped    | IMSI / MSISDN SLR                                                  |

Table 1

| Code | Message                                     | Description                                |
|------|---------------------------------------------|--------------------------------------------|
| 2001 | DIAMETER_SUCCESS                            | Success                                    |
| 5030 | DIAMETER_USER_UNKNOWN                       | OCS user not found<br>in Sy database       |
| 5570 | DIAMETER_ERROR_UNKNOWN_POLICY_COUNTERS      | Unknown policy counters<br>in OCS database |
| 4241 | DIAMETER_ERROR_NO_AVAILABLE_POLICY_COUNTERS | OCS no available policy counters           |

Code 5570 and 4241 are defined in TS 29.219 §5.5. Code 5030 is defined in RFC 4006.

Table 2

- PCRF - Gx and Rx interface Sy database
- AAA - HSS database
- AAA - HSS interface

# OmniHSS

□□□□□□

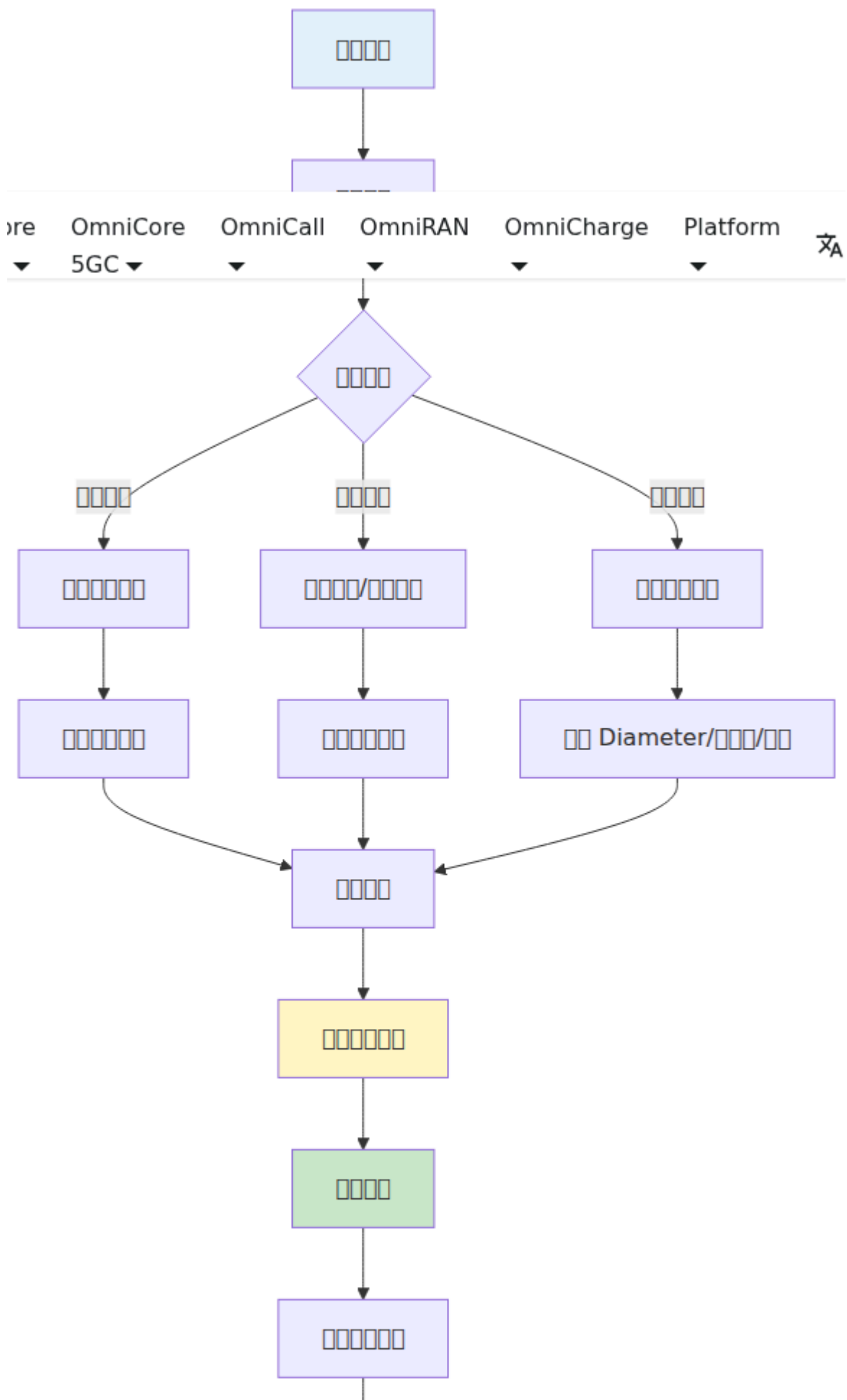
← □□□□□

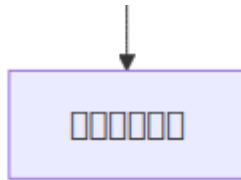
□□

- □□□□□
- □□□□
- Diameter □□□□
- □□□□□
- EPC □□□□
- IMS □□□□
- VoLTE □□□□
- □□□□
- EIR □□
- □□□□
- □□□□□□
- API □□
- □□□□□□□

□□□□□□

□□□□□□□□





□□□□□

□□□□□□□□□□□□□□□□

1. □□□□□□□□□□□□□□

- IMSI
- MSISDN□□□□□□
- □□□□□□
- □□□□□□□□

2. □□□□

- □□□□□□□□
- □□□□□□□□□□
- □□□□□□□□□□

3. □□□□

- □□□□□□□□□□
- □□□□□□□□□□
- □□□□□□□□/□□□□□□□□

4. □□□□

- □□ □□□□ □□□□□□□□
  - □□ Diameter □□□□□□
  - □□□□□□□□
-

# API

## API

- API
- "API" API
- API

## API

### API 1

#### API

- API
- API

#### API

1. API key\_set\_id

```
curl -k https://hss.example.com:8443/api/subscriber/imsi/[IMSI]
```

2. API

```
curl -k https://hss.example.com:8443/api/key_set/[KEY_SET_ID]
```

3. Ki OPC SIM

#### API

- API
- SIM

### API 2 SQN

#### API

- 000000000000
- 000“SQN 0000”
- 000000

000000

1. 000000000000 SQN 0
2. 000000 SQN 0000
3. 00000000 SQN 0

000000

- SQN 000000 AUTS 0000000000
- 0000000000000000 SQN 000 000000000000

000 00 SQN 000000000000000000000000

00 **3**00000000

000

- 00000000
- 0000000000

000000

1. 0000000000

```
curl -k https://hss.example.com:8443/api/subscriber/imsi/[IMSI]
```

2. 00 `enabled` 000 `true`

000000

- 000🔍🔍🔍🔍



```
curl -k -X PUT https://hss.example.com:8443/api/subscriber/[ID] \
-H "Content-Type: application/json" \
-d '{"subscriber": {"enabled": true}}'
```

## 4 EPC

### 

- 
- “ EPC ”

### 

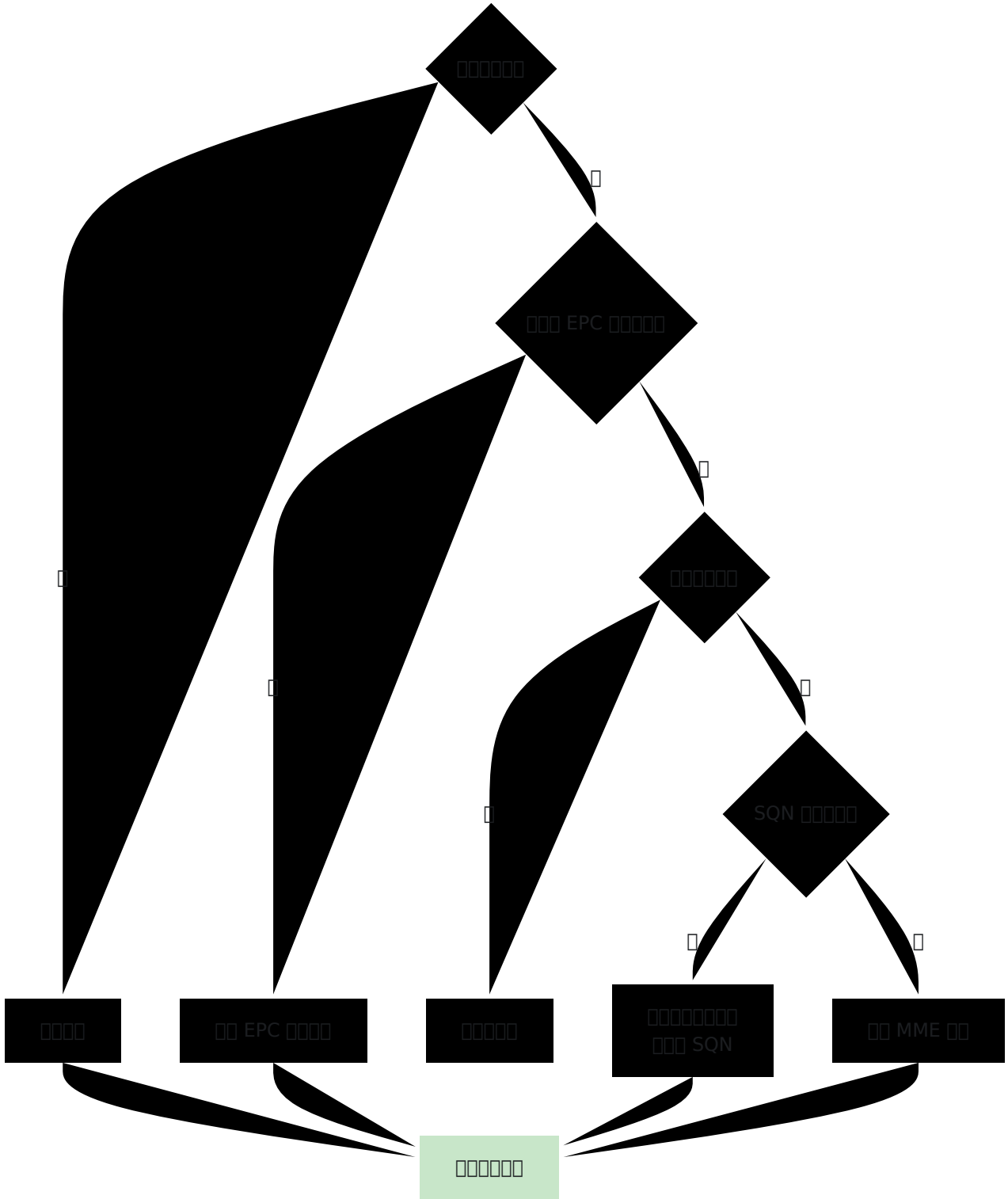
1. `epc_profile_id`
2. EPC

```
curl -k
https://hss.example.com:8443/api/epc/profile/[PROFILE_ID]
```

### 

- EPC

□□□□



# Diameter 检查

## 前提

- 在 网络 上 Diameter 服务器
- "网络" 是
- 网络

## 检查步骤

### 1. 检查

#### 检查

- 网络
- 网络
- Ping 网络

#### 网络

1. 检查 OmniHSS 网络

```
ping [PEER_IP]
```

2. 检查 Diameter 网络

```
telnet [PEER_IP] 3868
```

3. 检查 Diameter 网络 3868

#### 网络

- 网络
- 网络
- 网络

## 第2章 Diameter 配置

目标

- 配置
- CER/CEA 配置
- 配置

配置

1. 配置 runtime.exs 配置 Diameter 配置
  - 配置 origin\_host 配置
  - 配置 origin\_realm 配置
  - 配置 IP 配置
2. 配置 CER/CEA 配置
3. 配置 OmniHSS 配置 origin\_host

配置

- 配置 Diameter 配置 runtime.exs
- 配置 OmniHSS
- 配置

## 第3章 TLS Diameter

目标

- TLS 配置
- 配置
- "配置" 配置 "配置" 配置

配置

1. 配置 priv/cert/ 配置
2. 配置

```
openssl x509 -in priv/cert/diameter.crt -noout -dates
```

3. 設定

4. 設定 TLS

設定

- 設定
- 設定
- 設定 OmniHSS

4. 設定

設定

- 設定
- 設定
- "設定" 設定

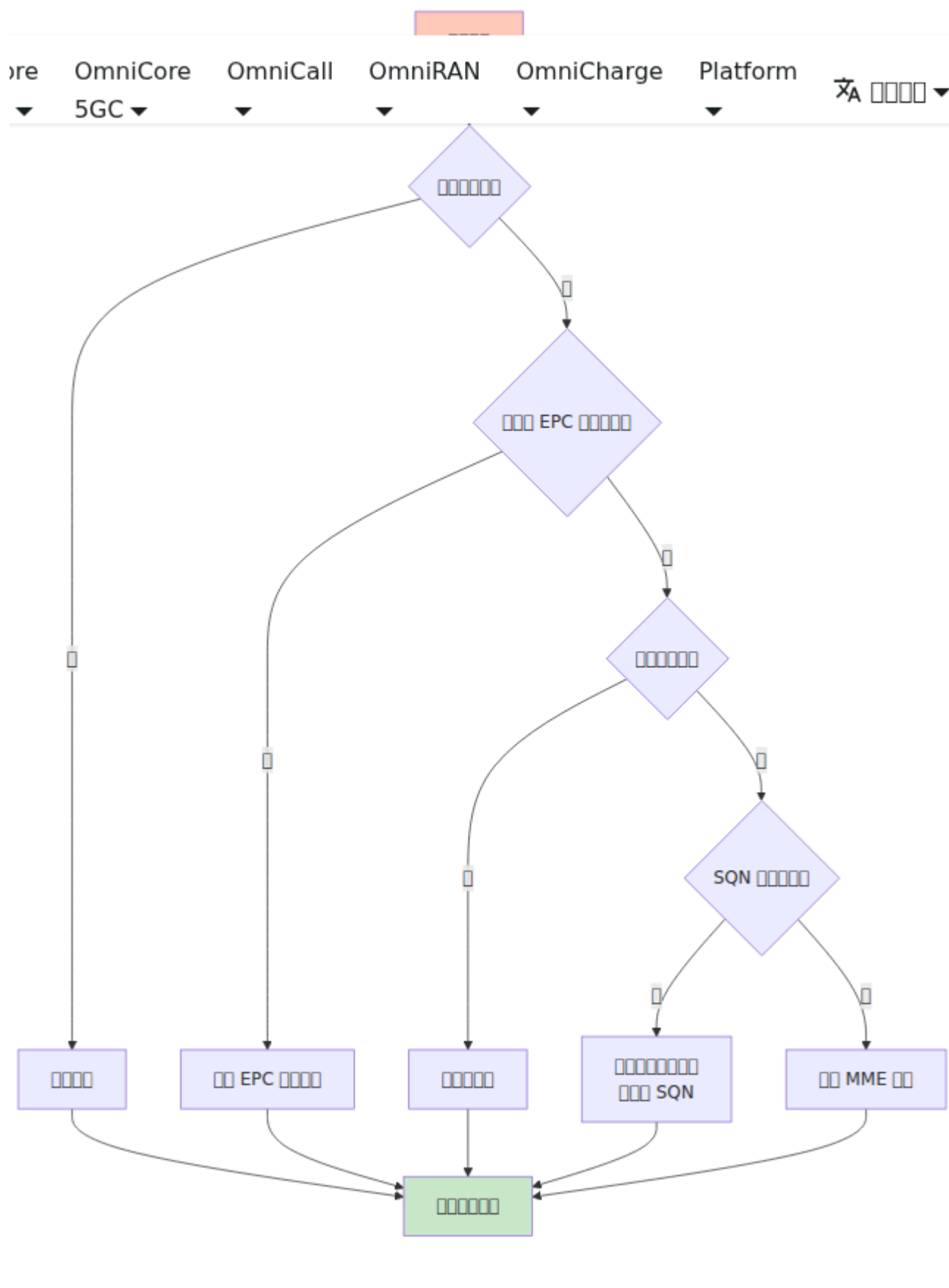
設定

1. 設定 Diameter 設定
2. 設定 S6a Cx Sh 設定
3. 設定 CER/CEA 設定

設定

- 設定 Diameter 設定
- 設定
  - MME 設定 S6a 16777251
  - S-CSCF 設定 Cx 16777216
  - P-GW 設定 Gx 16777238

# Diameter 网络架构



# 環境構築

## 前提

- API 500 行
- データベース
- "データベース" 名
- データベース

## 環境構築手順

### 1. データベース作成

#### 準備

- API 500 行
- データベース
- "データベース" 名

#### 実行

##### 1. データベース作成

```
データベース作成
psql -h [DB_HOST] -U [DB_USER] -d [DB_NAME]
```

##### 2. データベース接続確認

##### 3. データベース接続確認

#### 確認

- データベース
- データベース
- データベース

## 第 2 章 環境構築

### 前提

- "Visual Studio" のインストール
- OmniHSS のインストール

### インストール

1. runtime.exe のダウンロード
2. 実行環境のインストール
3. 実行環境の確認

### 確認

- runtime.exe の実行環境の確認
- 実行環境のインストール
- 実行環境の確認 OmniHSS

## 第 3 章 実行

### 前提

- 実行 500 回
- "Visual Studio" のインストール
- 実行環境の確認

### インストール

1. 実行環境のインストール
2. runtime.exe のダウンロード
3. 実行環境の確認

### 確認

- runtime.exe の実行環境の確認
- 実行環境のインストール
- 実行環境の確認



第 4 章

第 1 节

- API 接口
- 数据库
- CPU 性能

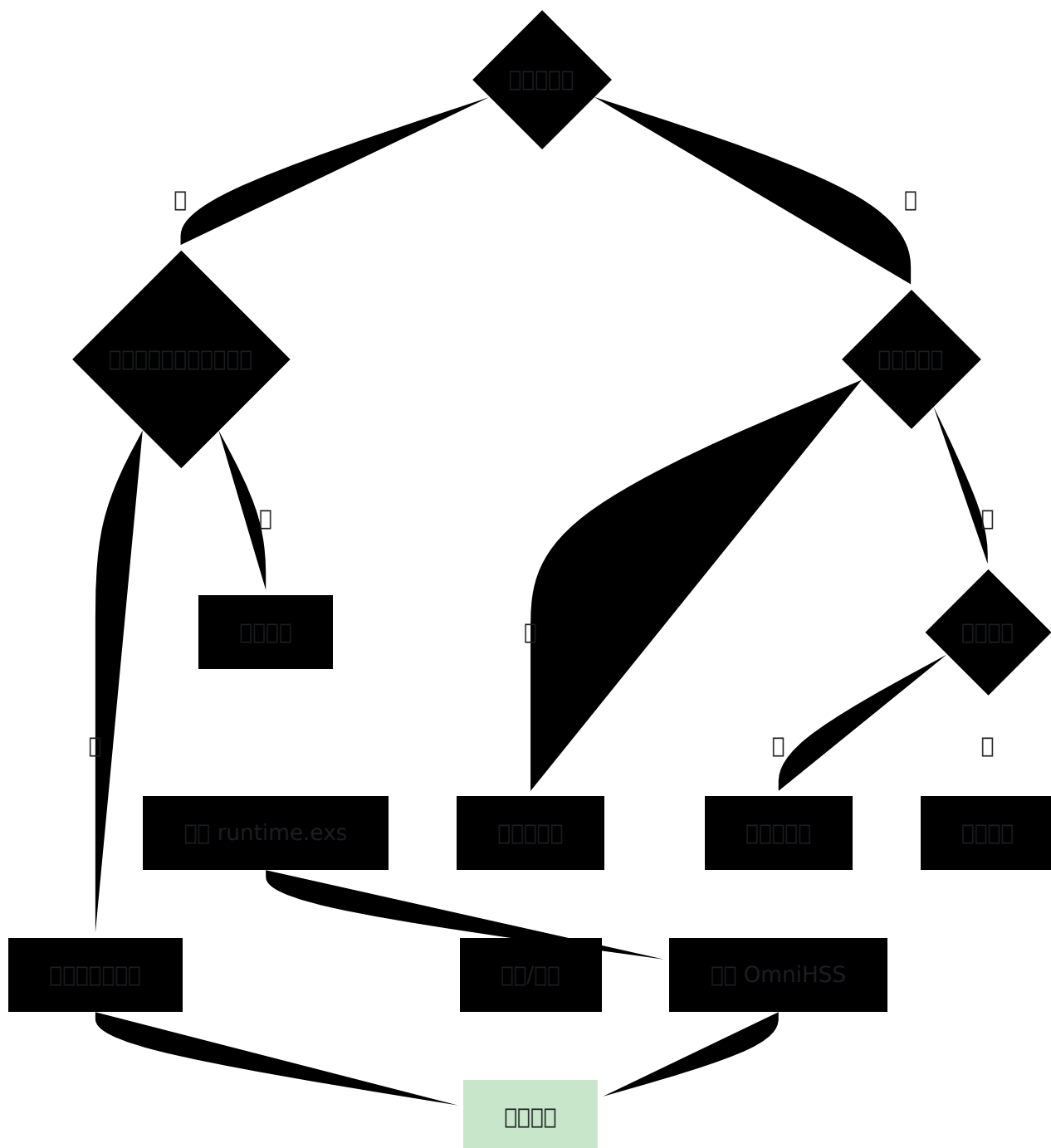
第 2 节

1. 数据库性能
2. 数据库性能
3. 数据库性能
4. 数据库性能

第 3 节

- 数据库
- 数据库
- 数据库
- 数据库

□□□□□



# EPC

## 

- 3GPP LTE
- MME
- PDN

## 

### 1

#### 

- 3GPP
- "3GPP" 3GPP
- 3GPP

#### 

1. roaming\_profile\_id
- 2.
3. MCC/MNC
- 4.

#### 

- MCC/MNC
- 
- 

### 2 APN

#### 

- PDN
- MME "APN"

- 網路存取點名稱

網路存取點

1. 網路 EPC 網路存取點名稱 APN 網路
2. 網路 APN 網路存取點名稱
3. 網路 APN 網路存取點

網路存取點

- 網路 APN 網路存取點 EPC 網路
- 網路 APN 網路存取點
- 網路 APN QoS 網路存取點

網路 3MME 網路

網路

- 網路存取點
- 網路 MME 網路
- Diameter 網路存取點

網路存取點

1. 網路 網路 Diameter 網路
2. 網路 MME 網路存取點 "網路"
3. 網路 MME 網路 S6a 網路

網路存取點

- 網路 Diameter 網路存取點
- 網路 MME 網路
- 網路 MME 網路

網路 4網路存取點

網路

- 網路存取點網路存取點

- 00000000
- 0000000000

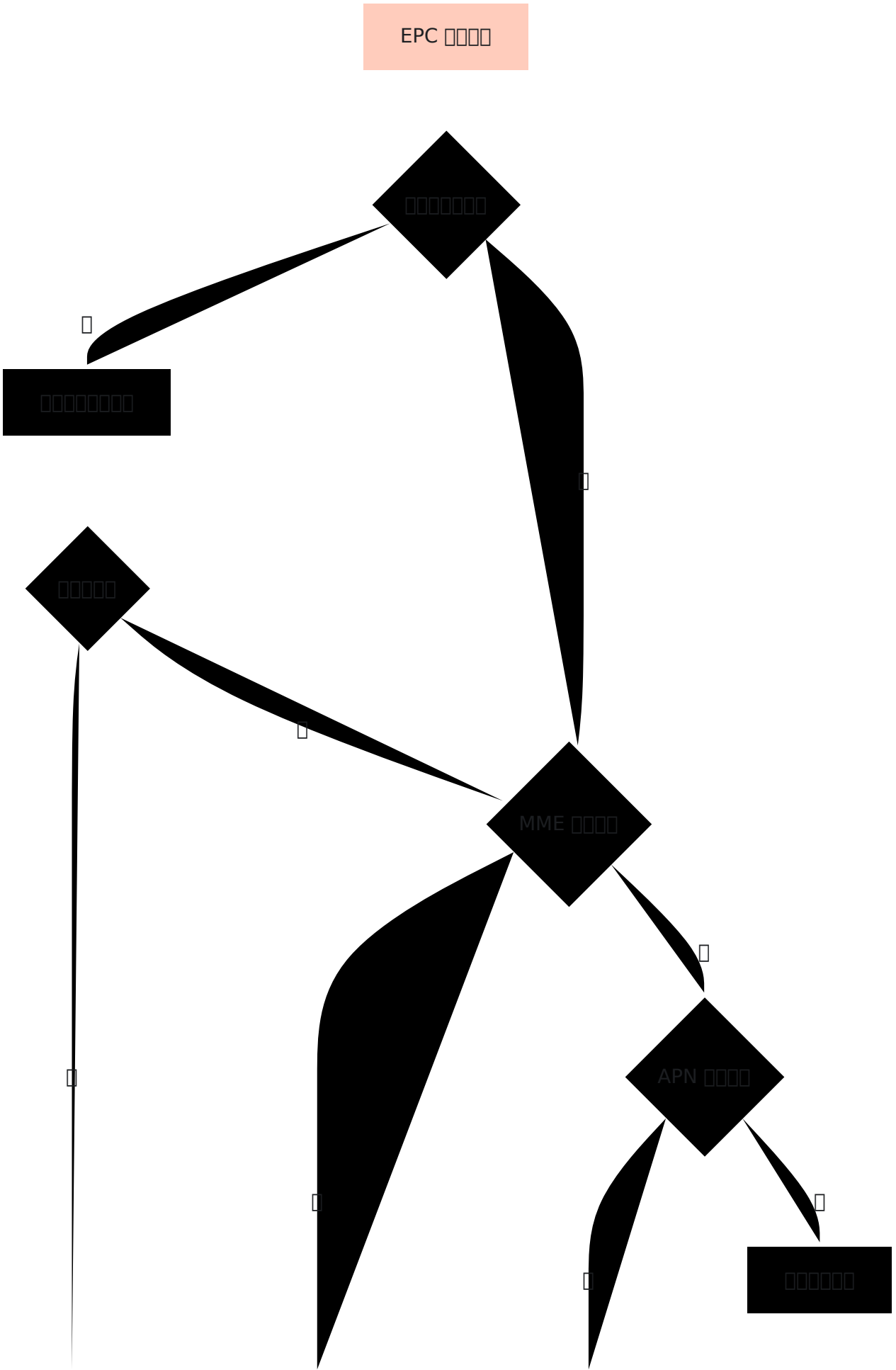
00000

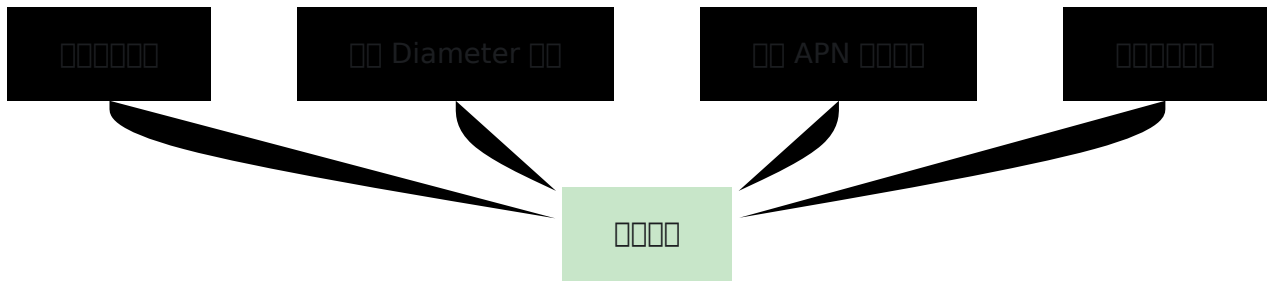
1. 0000000000
2. 00000 MME 00
3. 0000000000

00000

- 000000000000
- 0000000000 MME
- 00000000

**EPC** □□□□□□□□





## IMS

IMS

- IMS VoLTE
- IMS "IMS"
- IMS

IMS

IMS 1 IMS

IMS

- IMS
- IMS

IMS

- IMS `ims_enabled`
- IMS `ims_profile_id`

IMS

- IMS
- IMS

IMS 2 S-CSCF

IMS



- IMS 網元
- IMS 網元 Diameter 網元

網元

1. IMS 網元 Diameter 網元
2. IMS S-CSCF 網元
3. IMS S-CSCF 網元 Cx 網元

網元

- IMS S-CSCF 網元 Diameter 網元
- IMS S-CSCF 網元

IMS 3GPP 網元 IFC 網元

網元

- IMS 網元
- IMS 網元 IFC 網元

網元

1. IMS 網元
2. IMS IFC 網元
3. IMS IFC XML 網元

網元

- IMS 網元 IFC 網元 IMS 網元
- IMS 網元 IFC 網元

IMS 4GIMS 網元

網元

- IMS 網元
- IMS 網元
- IMS 網元 IMS 網元

□□□□

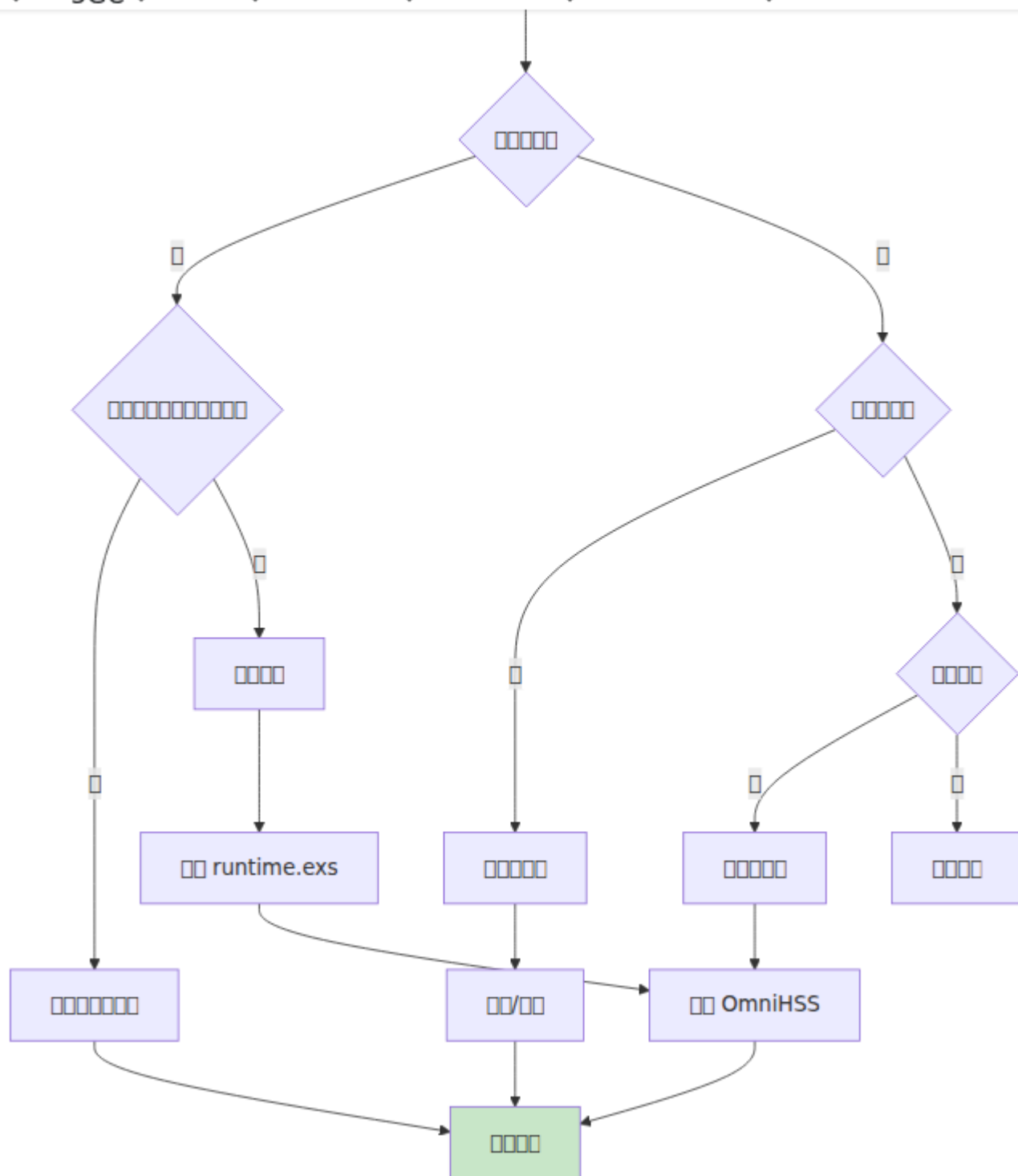
1. □□□□□□□ IMS □□
2. □□□□□□□□□□□□ □□ `ims_action`

□□□□

- □□ □□□□ □□ IMS
- □□□□□□□□□□□ IMS □□

**IMS** □□□□□□□□

Core 5G Core OmniCore OmniCall OmniRAN OmniCharge Platform    



# VoLTE 网络

网络

- IMS 网络
- 网络
- 网络
- 网络 "网络" 网络

网络网络网络

网络 1P-CSCF 网络

网络

- 网络网络
- 网络

网络

1. 网络 Diameter 网络
2. 网络 P-CSCF 网络
3. 网络 P-CSCF 网络 Rx 网络OmniHSS PCRF 网络

网络

- 网络 P-CSCF 网络 Diameter 网络
- 网络 P-CSCF 网络 OmniHSS 网络 Rx

网络 2网络

网络

- 网络网络
- AAR/AAA 网络
- Rx 网络

网络

1. 設定 Rx Diameter
2. 設定 AAR-AA-
3. 設定 AAA-AA-

- P-CSCF 設定 AAR
- OmniHSS Rx
- IMS

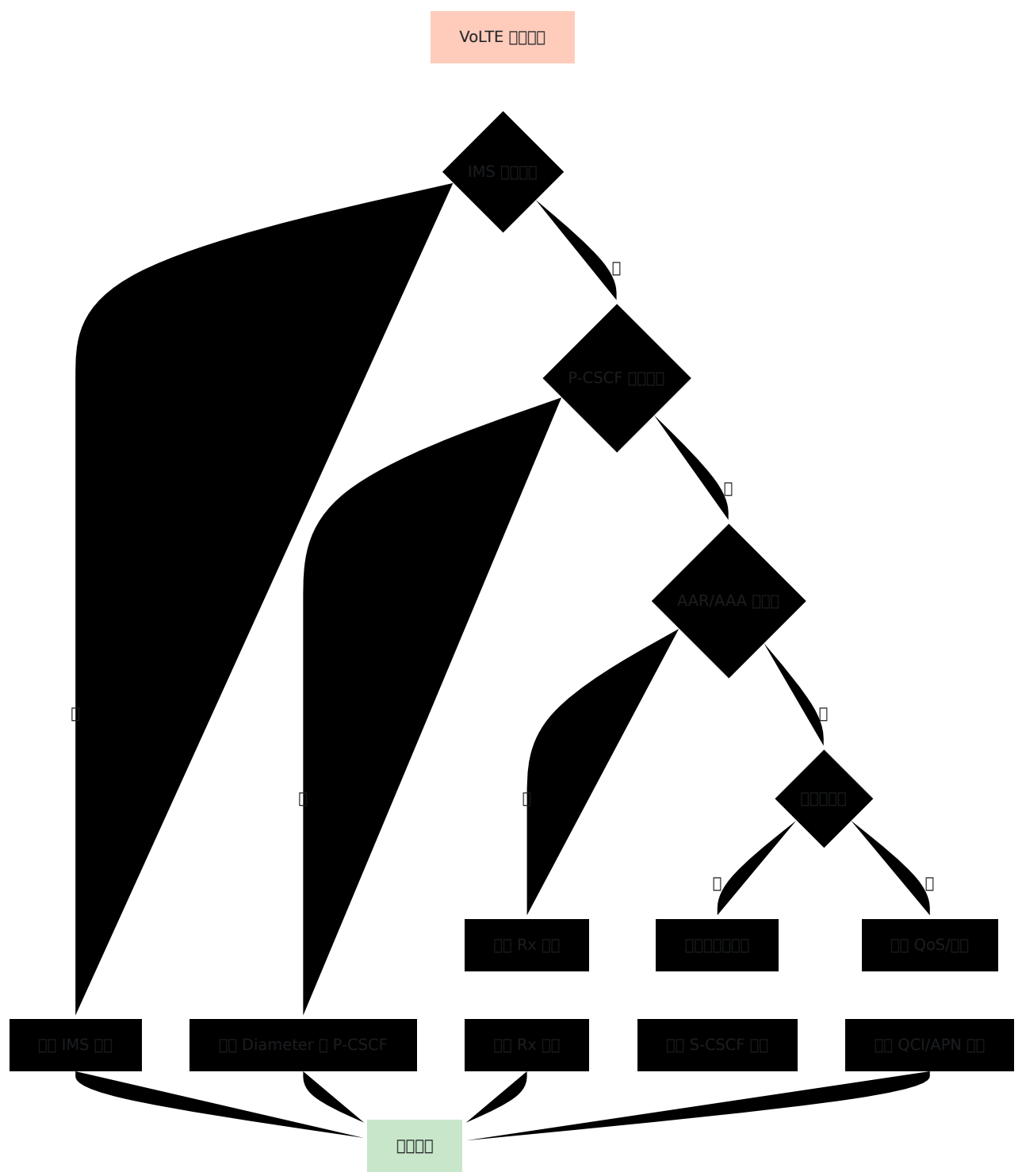
3QoS/

- 
- 
- 

1. APN APN QoS
2. QCI QCI 1
3. P-GW Gx-PCRF

- APN QoS IMS APN
- QCI 1
- P-GW Diameter

# VoLTE 网络架构



# AAA

## AA

- AAA
- AAA
- AAA

## AAAAA

### AA **1**AAAA

#### AAA

- AAA
- AAA

#### AAAA

1. AAA `roaming_profile_id`
2. AAA `null`

#### AAAA

- A AAA AAA

### AA **2**AAAA

#### AAA

- AAA
- AAA

#### AAAA

1. AAA MME AAA MCC/MNC
2. AAA
3. AAA MCC/MNC AAA

#### 4. 設定する

設定

- [API](#) 設定

```
curl -k -X POST https://hss.example.com:8443/api/roaming/rule \
-H "Content-Type: application/json" \
-d '{
 "roaming_rule": {
 "name": "Allow Visited Network",
 "mcc": "310",
 "mnc": "410",
 "data_action": "allow",
 "ims_action": "allow"
 }
}'
```

3 IMS

設定

- 設定
- IMS
- 設定

設定

1. 設定
2. `data_action` `ims_action`
3. 設定

設定

- IMS
  - `ims_action: "allow"`
- `ims_action_if_no_rules_match` `"allow"`

[API](#) 設定

---



# EIR 查詢

## 查詢

- 查詢號碼
- 查詢號碼
- EIR 查詢

## 查詢號碼

查詢 1 查詢 IMEI 查詢

## 查詢

- 查詢號碼/查詢
- 查詢號碼

## 查詢

- 查詢 EIR 查詢
- 查詢號碼
- 查詢號碼 IMEI
- 查詢號碼/查詢

## 查詢

- 查詢號碼 EIR 查詢
- 查詢號碼
- 查詢號碼

查詢 2 MME 查詢 S13 查詢

## 查詢

- EIR 查詢
- 查詢號碼

## 查詢

1. 查詢 MME 資料庫 S13 查詢
2. 查詢 MME Diameter 資料庫
3. 查詢 S13 資料庫
4. 查詢 MME 查詢

查詢結果

- 查詢 MME 查詢 S13 查詢 EIR 查詢
- 查詢 Diameter 資料庫 S13 查詢 (16777252)
- 查詢資料庫 MME 查詢

查詢 3 查詢結果

查詢

- 查詢資料庫查詢結果

查詢結果

1. 查詢 EIR 查詢
2. 查詢資料庫查詢結果
3. 查詢結果

查詢結果

- 查詢資料庫 . \* 查詢結果 IMEI
- 查詢資料庫查詢結果
- 查詢資料庫查詢結果

---

查詢結果

查詢

- API 查詢
- Diameter 查詢
- CPU 查詢結果

- 00000000

00000000

00 **1**0000000

000

- 000000
- 000 CPU 0
- 0000

00000

1. 000000000000000
2. 00000
3. 0000000
4. 000000

00000

- 00000
- 0000000
- 0000000
- 0000000
- 000 00000

00 **2**000000

0 0 0

- 000 0 0 000
- 000000000
- 0000000

00000

1. 000000

2. 鋼管
3. 鋼管管口
4. 鋼管管口

鋼管

- 鋼管
- 鋼管管口
- 鋼管管口
- 鋼管管口

### 3 Diameter 鋼管

鋼管

- Diameter 鋼管
- 鋼管管口
- 鋼管管口

鋼管

1. 鋼管 Diameter 鋼管
2. 鋼管管口
3. 鋼管管口
4. 鋼管管口

鋼管

- 鋼管管口
- 鋼管管口
- 鋼管管口
- 鋼管管口 Diameter 鋼管

### 4 鋼管

鋼管

- OmniHSS 鋼管

- 0000000
- 00000000

000000

1. 00000000 OmniHSS 0000000
2. 0000000
3. 0000000
4. 00 Erlang VM 00

000000

- 00 OmniHSS 00000000
- 00000000000000000000
- 0 runtime.exs 000 Erlang VM 0000
- 0000000000000000

0000000??

00

- 0000000000000000
- 0000000
- 00000000
- 0000000

00000000000

00 10MME 00/00

000

- 0000000 MME 00 MME 0000000
- 000 MME 00000000
- 0000

□□□□

1. □□□□□□□□ MME
2. □□ MME □□□□
3. □□ MME □□□□□□

□□□□

- □□□□□□□□□□□□
- □□□□□□□□
- MME □□□□□□ Cancel-Location

□□ **2**□□□□□□□□

□□□

- □□□□□□□□□□□□
- PDN □□□□□□□□□□
- □□□□□

□□□□

1. □□□□ last\_seen □□□
2. □□□□□□□□□□□□□□
3. □□□□□□□□□□

□□□□

- □□□□□□□□□□□□□□
- □□□□□□□□□□□□□□
- □□□□□□□□□□□□□□□□

□□ **3**□□□□□□

□□□

- □□□□□□□□
- □□□□
- □□□□□

□□□□

1. □□□□□□□□□□
2. □□□□□□
3. □□□□□□

□□□□

- □□□□□□□□□□
- □□□□□□□□□□
- □□□□□□□□□□

---

# API □□

□□

- API □□□□
- API □□□□
- □□□□/□□□□
- 500 □□

□□□□□□□□□□

□□ **1**□□□□□□□□

□□□

- 400 □ 422 □□
- □□□□□□
- □□□□□

□□□□□

1. □□□□□□□□□□□□□□
2. □□ API □□□□

3. 認証情報取得

4. 認証完了

認証完了

- 認証情報取得 API 呼び出し
- 認証情報取得
- 認証情報取得 ID 取得

第 2 段階

認証

- 認証情報取得
- 認証情報取得 “key\_set\_id 取得”
- 認証情報取得

認証完了

1. 認証情報取得
2. 認証情報取得
  - key\_set\_id → 取得
  - epc\_profile\_id → EPC 取得
  - ims\_profile\_id → IMS 取得

認証完了

- 認証情報取得
- 認証情報取得 ID
- 認証情報取得

第 3 段階

認証

- 500 取得
- API 取得
- 認証情報取得



□□□□

- □□ □□□□

□□□□□□□□

□□□□□□□□

1. □□□□

- URL: `https://[hostname]:7443/overview`
- □□□□□□□□□□□□□□□□

2. **Diameter** □□

- URL: `https://[hostname]:7443/diameter`
- □□□□□□□□□□□□□□

3. □□□□

- URL: `https://[hostname]:7443/application`
- □□□□□□□□□□□□□□□□

**API** □□□□

□□□□□□□□

```
curl -k https://hss.example.com:8443/api/status
```

□□□□□□

```
IMSI
curl -k https://hss.example.com:8443/api/subscriber/imsi/001001123456789

MSISDN
curl -k https://hss.example.com:8443/api/subscriber/msisdn/14155551234

ID
curl -k https://hss.example.com:8443/api/subscriber/d290f1ee-6c54-4b01-90e6-d701748f0851
```

□□□□□□

```
curl -k https://hss.example.com:8443/api/subscriber
```

□□□□□□□□

```
EPC □□□□
curl -k https://hss.example.com:8443/api/epc/profile/e58ed763-928c-4155-bee9-fdbaaadc15f3

IMS □□□□
curl -k https://hss.example.com:8443/api/ims/profile/7c9e6679-7425-40de-944b-e07fc1f90ae7

□□□□□□
curl -k https://hss.example.com:8443/api/roaming/profile/9c858901-8a57-4791-81fe-4c455b099bc9
```

□□□□□□

□□ **Diameter** □□□□□□

```
telnet [PEER_IP] 3868
```

□□ **TLS** □□□

```
openssl s_client -connect [hostname]:8443 -showcerts
```

□□□□□□□□

```
psql -h [DB_HOST] -U [DB_USER] -d [DB_NAME] -c "SELECT COUNT(*)
FROM subscriber;"
```

□□□□

□□□□□□□□ **IMSI**□

```
grep "001001123456789" /var/log/omnihss/omnihss.log
```

□□□□□□□□

```
grep "authentication.*fail" /var/log/omnihss/omnihss.log
```

□□ **Diameter** □□□□□□

```
grep "Diameter peer" /var/log/omnihss/omnihss.log
```

□□□□□□□□

```
grep -i "database.*error" /var/log/omnihss/omnihss.log
```

---

□□□□

□□□□

□□□□□□□□□□□□□□□□/□□□□□□

1.
2.
3.
4.
5.
6.

|  |  |  |  |  |
|--|--|--|--|--|
|  |  |  |  |  |
|--|--|--|--|--|

□□□□□□□□

1. **OSINT** - Open Source Intelligence
2. **OSINT** - Open Source Intelligence
3. **OSINT** - Open Source Intelligence
4. **OSINT** - runtime.exe Open Source Intelligence
5. **OSINT** - OmniHSS Open Source Intelligence
6. **OSINT** - Open Source Intelligence
7. **OSINT** - Open Source Intelligence IMSI

|  |  |  |  |  |  |
|--|--|--|--|--|--|
|  |  |  |  |  |  |
|--|--|--|--|--|--|

□□□□□□□□□□

- 
- 
- 
- 

[illegible]

- 
- 
- 
- 

□□□□□□□□

□□□□□□

| □□□□         | □□            | □□□□         |
|--------------|---------------|--------------|
| "□□□□□□□□□□" | □□□□□□□□□□    | □□ □□□□□     |
| "SQN □□□□"   | SQN □□□       | □□□□□□       |
| "□□□□□"      | □□□ IMSI      | □□ IMSI□□□□□ |
| "□□□□□"      | enabled=false | □□□□         |

Diameter □□

| □□□□                | □□         | □□□□           |
|---------------------|------------|----------------|
| "Diameter □□□□□□□□" | □□□□       | □□□□□□         |
| "CER/CEA □□□□"      | □□□□□      | □□ Diameter □□ |
| "□□□□□□□□"          | □□□□□□□□□□ | □□□□□□□        |
| "TLS □□□□"          | □□□□       | □□□□           |

□ □ □ □ □

|          |      |       |
|----------|------|-------|
| □□□□     | □□   | □□□□  |
| "□□□□"   | □□□□ | □□□□  |
| "□□□□□"  | □□□□ | □□□□  |
| "□□□□□□" | □□□  | □□□□□ |
| "□□□□"   | □□□  | □□□□  |

**API** ☐ ☐

| 項目名             | 項目      | 項目値             |
|-----------------|---------|-----------------|
| "key_set_id 項目" | 項目値     | 項目値項目値          |
| "IMSI 項目"       | 項目 IMSI | 項目値 IMSI 項目値項目値 |
| "項目"            | 項目      | 項目値項目値項目値       |

# OmniHSS Webhook 簡介

← 目錄

---

## 簡介

- 簡介
  - Webhook 簡介
  - Webhook 類型
  - Webhook 格式
  - 配置
  - 驗證
  - 錯誤處理
  - 常見問題
- 

## 簡介

OmniHSS 提供 **webhooks** 功能，用於接收來自 IMS 系統的實時事件通知。OmniHSS 系統通過 webhooks 發送 HTTP POST 請求，通知客戶端系統。

## Webhooks

Webhooks 是 HTTP 協議的一部分，OmniHSS 系統通過 HSS API 接口發送。





概要

- 概要 - Webhooks の概要
- 概要 - Webhook の 5 種類
- 概要 - 各 Webhook の概要
- 概要 - 各 Webhook の URL

概要 各 Webhook の概要 Webhook の URL Webhook の URL

# Webhook 概要

OmniHSS 各 Webhooks

## EPC/LTE 概要

| Webhook                            | API     | 概要               |
|------------------------------------|---------|------------------|
| update_location_request            | S6a ULR | 位置情報更新リクエスト      |
| authentication_information_request | S6a AIR | 認証情報リクエスト        |
| purge_request                      | S6a PUR | MME の位置情報削除リクエスト |
| cancel_location_answer             | S6a CLA | MME の位置情報取消回答    |

# IMS

| API                 | Interface       | Function               |
|---------------------|-----------------|------------------------|
| ims_registration    | Cx SAR          | IMS/VoLTE registration |
| ims_deregistration  | Cx SAR (de-reg) | IMS deregistration     |
| ims_profile_request | Sh UDR          | IMS profile request    |

## PCRF (PCRF)

| API                 | Interface | Function                       |
|---------------------|-----------|--------------------------------|
| policy_request      | Gx CCR    | P-GW policy request            |
| media_authorization | Rx AAR    | P-CSCF IMS media authorization |

## IMSI

| API         | Interface                          | Function        |
|-------------|------------------------------------|-----------------|
| imsi_switch | ULR for different IMSI on same SIM | IMSI SIM switch |

# Webhook

## Webhook

OmniHSS webhook URL HTTP POST

POST /your-webhook-endpoint HTTP/1.1

Host: your-server.com

Content-Type: application/json

X-OmniHSS-Event: update\_location\_request

X-OmniHSS-Event-ID: 550e8400-e29b-41d4-a716-446655440000

X-OmniHSS-Timestamp: 2025-01-15T14:30:00Z

```
{
 "event": "update_location_request",
 "event_id": "550e8400-e29b-41d4-a716-446655440000",
 "timestamp": "2025-01-15T14:30:00Z",
 "subscriber": {
 "id": "d290f1ee-6c54-4b01-90e6-d701748f0851",
 "imsi": "001001123456789",
 "enabled": true,
 "ims_enabled": true,
 "msisdns": [
 {"id": "16fd2706-8baf-433b-82eb-8c7fada847da", "msisdn":
"14155551001"},
 {"id": "27ae3807-9cbf-4a4c-93fc-9d8afdb958eb", "msisdn":
"14155551002"}
],
 "sim": {
 "id": "3f2504e0-4f89-41d3-9a0c-0305e82c3301",
 "iccid": "8991101200003204510",
 "is_esim": false
 },
 "key_set": {
 "id": "6ba7b810-9dad-41d1-80b4-00c04fd430c8",
 "amf": "8000"
 },
 "epc_profile": {
 "id": "e58ed763-928c-4155-bee9-fdbaaadc15f3",
 "name": "Premium 100Mbps",
 "ue_ambr_dl_kbps": 100000,
 "ue_ambr_ul_kbps": 50000
 },
 "ims_profile": {
 "id": "7c9e6679-7425-40de-944b-e07fc1f90ae7",
 "name": "Standard VoLTE"
 },
 "roaming_profile": {
 "id": "9c858901-8a57-4791-81fe-4c455b099bc9",
```

```
 "name": "International Roaming Allowed"
 },
 "subscriber_state": {
 "mme_host": "mme-01.example.com",
 "mme_realm": "epc.mnc001.mcc001.3gppnetwork.org",
 "visited_plmn": "001001",
 "last_update": "2025-01-15T14:30:00Z"
 },
 "custom_attributes": {
 "account_type": "premium",
 "billing_plan": "unlimited"
 }
},
"event_context": {
 "visited_plmn": "310410",
 "mme_host": "mme-roaming.example.com",
 "location_update_type": "initial_attach"
}
}
```

Schema

| Field         | Type   | Description                                           |
|---------------|--------|-------------------------------------------------------|
| event         | string | Event name, e.g., update_location_request             |
| event_id      | string | Unique identifier for the event, e.g., webhook UUID   |
| timestamp     | string | Event timestamp in ISO 8601 format                    |
| subscriber    | object | Subscriber information, e.g., GET /api/subscriber/:id |
| event_context | object | Event context information                             |

Example

event\_context {

update\_location\_request:

```
{
 "visited_plmn": "310410",
 "mme_host": "mme-roaming.example.com",
 "mme_realm": "epc.mnc410.mcc310.3gppnetwork.org",
 "location_update_type": "initial_attach"
}
```

## □□ **imsi\_switch:**

```
{
 "previous_imsi": "001001111111111",
 "new_imsi": "310410222222222",
 "sim_id": "3f2504e0-4f89-41d3-9a0c-0305e82c3301",
 "previous_mme_host": "mme-home.example.com",
 "new_mme_host": "mme-roaming.example.com"
}
```

## □□ **ims\_registration:**

```
{
 "scscf_host": "scscf-01.ims.example.com",
 "public_identities": [
 "sip:001001123456789@ims.mnc001.mcc001.3gppnetwork.org",
 "sip:+14155551001@ims.example.com",
 "tel:+14155551001"
]
}
```

# HTTP

| Header              | Value                   | Value                   |
|---------------------|-------------------------|-------------------------|
| Content-Type        | application/json        | application/json        |
| X-OmniHSS-Event     | update_location_request | update_location_request |
| X-OmniHSS-Event-ID  | UUID                    | UUID                    |
| X-OmniHSS-Timestamp | ISO 8601                | ISO 8601                |
| User-Agent          | OmniHSS                 | OmniHSS/1.0             |

## Webhooks

Webhooks are used to receive data from the OmniHSS API.

### Webhook

```
curl -k -X POST https://hss.example.com:8443/api/webhook \
-H "Content-Type: application/json" \
-d '{
 "webhook": {
 "url": "https://your-server.com/omnihss-webhook",
 "events": [
 "update_location_request",
 "ims_registration",
 "imsi_switch"
],
 "enabled": true,
 "description": "omnihss webhook"
 }
}'
```

📄:

```
{
 "data": {
 "id": "4e5d6c7b-8a9f-4b0c-8d1e-2f3a4b5c6d7e",
 "url": "https://your-server.com/omnihss-webhook",
 "events": [
 "update_location_request",
 "ims_registration",
 "imsi_switch"
],
 "enabled": true,
 "description": "omnihss webhook",
 "created_at": "2025-01-15T14:00:00Z"
 }
}
```

## 📄 Webhooks

```
curl -k https://hss.example.com:8443/api/webhook
```

## 📄 Webhook

```
curl -k -X PUT https://hss.example.com:8443/api/webhook/4e5d6c7b-8a9f-4b0c-8d1e-2f3a4b5c6d7e \
-H "Content-Type: application/json" \
-d '{
 "webhook": {
 "enabled": false
 }
}'
```

## Webhook

```
curl -k -X DELETE
https://hss.example.com:8443/api/webhook/4e5d6c7b-8a9f-4b0c-8d1e-2f3a4b5c6d7e
```

## Webhook

webhook

1. **POST** Content-Type: application/json
2. - 5 HTTP 200-299
3. -
4. **HTTPS** - TLS/SSL
5. - OmniHSS

**Webhook (Node.js/Express):**



```

const express = require('express');
const app = express();

app.post('/omnihss-webhook', express.json(), (req, res) => {
 const { event, subscriber, event_context } = req.body;

 console.log(`📡: ${event}`);
 console.log(`📡 IMSI: ${subscriber.imsi}`);

 // 📡 📡 📡 📡 📡
 // ... 📡 📡 📡 📡 📡 ...

 // 📡 📡 📡 📡 📡
 res.status(200).json({ received: true });

 // 📡 📡 📡 📡 📡
 processWebhook(req.body).catch(console.error);
});

async function processWebhook(payload) {
 // 📡 📡 📡 📡 📡 📡
 // 📡 📡 📡 📡 📡 📡 📡 📡 📡 📡 📡 📡
}

app.listen(3000);

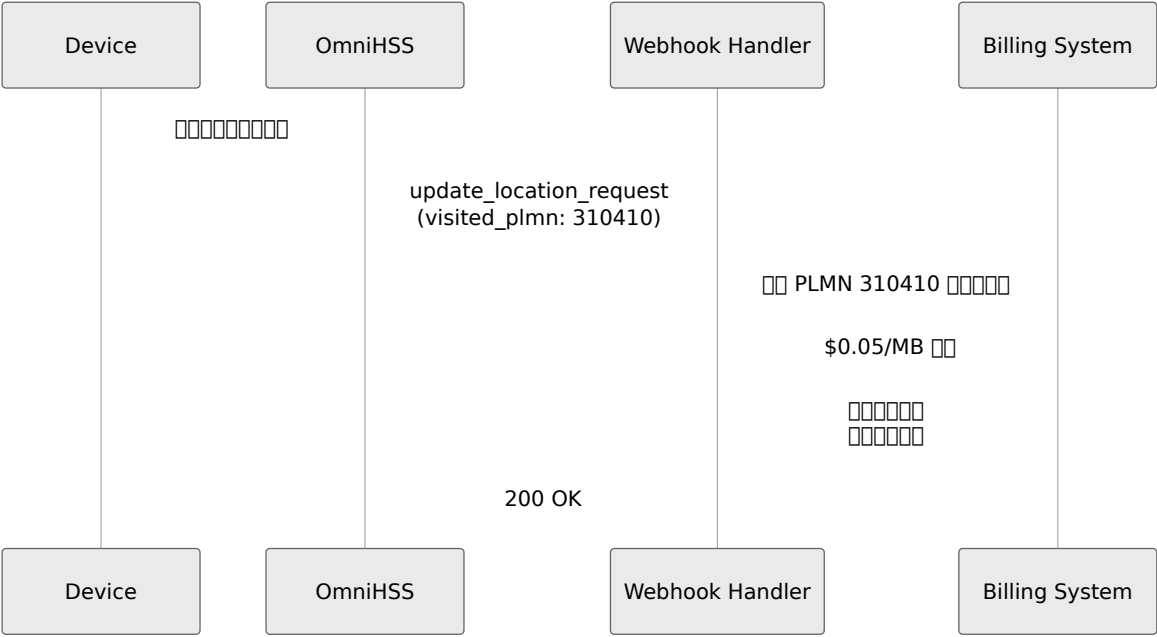
```

---

📡

## 1. 📡 📡 📡 📡 📡 📡

📡 📡 📡 📡 📡 📡 📡 📡 📡 📡 📡 📡



：

- 
- 
- 
- 

2.

：

```
// Webhook 接收事件通知
app.post('/omnihss-webhook', async (req, res) => {
 const { event, subscriber, event_context } = req.body;

 if (event === 'update_location_request') {
 await analytics.track({
 event: 'subscriber_location_update',
 imsi: subscriber.imsi,
 visited_plmn: event_context.visited_plmn,
 timestamp: req.body.timestamp,
 profile: subscriber.epc_profile.name
 });
 }

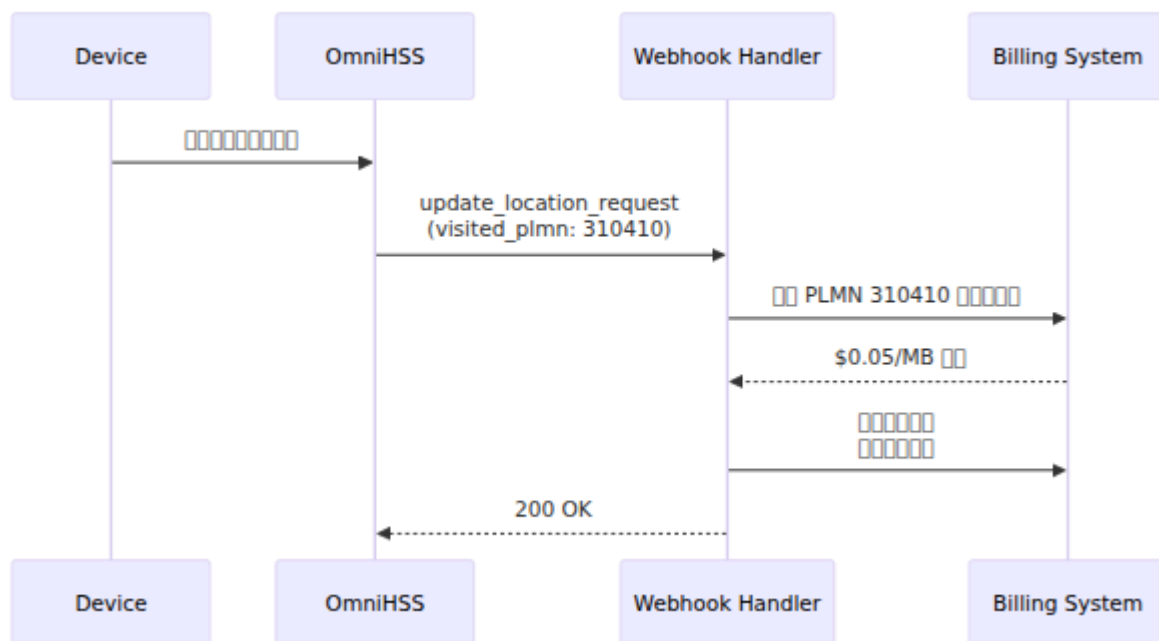
 res.status(200).send();
});
```

注意：

- MME 接收事件
- 接收事件
- 接收事件
- IMS 接收事件

### 3. 接收事件

接收事件



Sequence:

### 1. Request

- User A
- 30 User B
- :

### 2. IMSI

- SIM IMSI
- SIM IMSI
- : SIM IMSI

### 3. Response

- 
- :

Sequence:

```

@app.route('/omnihss-webhook', methods=['POST'])
def webhook_handler():
 data = request.json
 subscriber = data['subscriber']
 event_context = data.get('event_context', {})

 if data['event'] == 'update_location_request':
 visited_plmn = event_context.get('visited_plmn')

 # 黑名单检查
 if visited_plmn in BLOCKED_PLMNS:
 disable_subscriber(subscriber['imsi'])
 alert_security_team(subscriber, '黑名单 PLMN')

 # 不可能旅行检查
 if is_impossible_travel(subscriber['imsi'], visited_plmn):
 flag_for_review(subscriber['imsi'])
 alert_fraud_team(subscriber, '不可能旅行')

 return jsonify({'status': 'ok'}), 200

```

## 4. 部署

部署环境要求如下：

OS: Ubuntu 18.04 LTS VolTE 支持 IMS

```

app.post('/omnihss-webhook', async (req, res) => {
 const { event, subscriber } = req.body;

 if (event === 'ims_registration' && !subscriber.ims_enabled) {
 // IMS 注册 - 启用 IMS
 await omnihss.updateSubscriber(subscriber.id, {
 ims_enabled: true,
 custom_attributes: {
 ...subscriber.custom_attributes,
 volte_activated_at: new Date().toISOString()
 }
 });

 // 更新 CRM
 await crm.updateCustomer(subscriber.imsi, {
 features: ['volte']
 });
 }

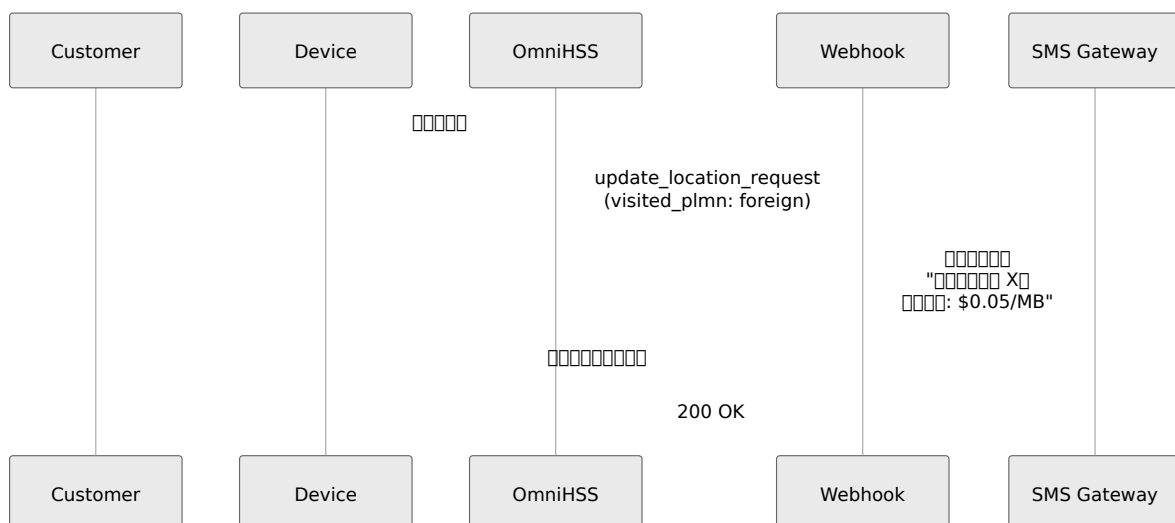
 res.status(200).send();
});

```

## 5. 测试

测试环境部署完成

测试: 测试环境部署完成



测试:

- "IMEI [IMEI] is not supported"
- "IMEI 80% is not supported"
- "IMEI is not supported VoLTE is not supported"
- "IMEI is not supported"

## 6. IMSI SIM

IMSI SIM is a feature that allows a user to switch their SIM card without changing their IMSI number.

```
app.post('/omnihss-webhook', async (req, res) => {
 const { event, subscriber, event_context } = req.body;

 if (event === 'imsi_switch') {
 const { previous_imsi, new_imsi, sim_id } = event_context;

 // Log IMSI switch
 await db.logImsiSwitch({
 sim_id,
 from_imsi: previous_imsi,
 to_imsi: new_imsi,
 timestamp: req.body.timestamp
 });

 // End session
 await billing.endSession(previous_imsi);
 await billing.startSession(new_imsi);

 // Check if switch count is high
 const switchCount = await db.getSwitchCount(sim_id, '24h');
 if (switchCount > 10) {
 await alertFraudTeam(`IMSI switch: SIM ${sim_id}`);
 }
 }

 res.status(200).send();
});
```

## 7. OmniHSS

OmniHSS is a feature that allows a user to switch their SIM card without changing their IMSI number.

📌:

- **CRM** 📌 - 📌📌📌📌📌📌📌📌
- 📌📌📌 - 📌📌📌📌📌📌📌📌
- 📌📌📌 - 📌📌📌📌📌📌
- 📌📌📌 - 📌📌📌📌📌📌📌📌
- 📌📌📌 - 📌📌📌📌📌📌📌📌📌📌

📌📌📌📌

## Webhook 📌/📌

📌📌📌 webhooks 📌 OmniHSS📌📌📌📌📌📌

```
📌📌📌📌📌 webhook
curl -k -X POST https://hss.example.com:8443/api/webhook \
-H "Content-Type: application/json" \
-d '{
 "webhook": {
 "url": "https://your-server.com/omnihss-webhook",
 "events": ["update_location_request"],
 "secret": "your-secret-key-here"
 }
}'
```

OmniHSS 📌📌📌📌 X-0mniHSS-Signature 📌📌

X-0mniHSS-Signature:  
sha256=5d7a8f9b2c1e3a4d6f7e8b9c0a1b2c3d4e5f6a7b8c9d0e1f2a3b4c5d6e7f8a

📌📌📌:



```

const crypto = require('crypto');

function verifyWebhook(req) {
 const signature = req.headers['x-omnihss-signature'];
 const secret = process.env.WEBHOOK_SECRET;
 const payload = JSON.stringify(req.body);

 const expectedSignature = 'sha256=' +
 crypto.createHmac('sha256', secret)
 .update(payload)
 .digest('hex');

 return crypto.timingSafeEqual(
 Buffer.from(signature),
 Buffer.from(expectedSignature)
);
}

app.post('/omnihss-webhook', (req, res) => {
 if (!verifyWebhook(req)) {
 return res.status(401).json({ error: 'invalid' });
 }

 // OK webhook...
 res.status(200).send();
});

```

## 总结

1. 使用 **HTTPS** - 使用 webhook 使用 TLS
2. 使用 - 使用 webhook 使用
3. 使用 - 使用 webhook 使用
4. **IP** 使用 - 使用 webhook 使用 OmniHSS IP 使用
5. 使用 - 使用 webhook 使用
6. 使用 - 使用 webhook 使用
7. 使用 - 使用 webhook 使用

## Webhook

Webhook 支持以下事件

- IMSI 注册/注销
- MSISDN 注册/注销
- 网络切片 PLMN 变更
- 网络切片选择

Webhook 支持以下配置

- **GDPR** - 通过 webhook 接收 GDPR 数据
- 鉴权 - 通过 webhook 接收鉴权数据
- 网络切片 - 通过 webhook 接收网络切片数据
- 安全 - 通过 webhook 接收 TLS 数据
- 网络切片 - 通过 webhook 接收网络切片数据

## Webhook 配置

### Webhook 配置

配置:

- 通过 webhook 接收数据
- Webhook 接收鉴权数据

配置示例:

#### 1. 通过 webhook 接收数据:

```
curl -k https://hss.example.com:8443/api/webhook
设置 "enabled": true
```

#### 2. 通过 webhook 接收鉴权:

- 設定する webhook の events 指定
- 例: 指定する `ims_registration` のイベントを監視する

### 3. HSS 設定

- webhook 設定
- 設定項目
- DNS 設定

### 4. テスト

```
curl -X POST https://your-server.com/omnihss-webhook \
-H "Content-Type: application/json" \
-d '{"test": true}'
```

## Webhook 概要

概要

- HSS からの webhook 受信
- Webhook 受信後 HSS からの応答

フロー

#### 1. 受信

- 5 秒以内 HTTP 200
- 応答内容

#### 2. 送信

```
// 10 - 10초 지연
app.post('/webhook', (req, res) => {
 processData(req.body); // 10초 지연
 res.status(200).send();
});

// 20 - 20초 지연
app.post('/webhook', (req, res) => {
 res.status(200).send(); // 20초 지연
 processData(req.body); // 20초 지연
});
```

## Webhooks

Webhooks

- OmniHSS에서 발생하는 이벤트
- 이벤트 ID (event\_id)를 포함

Webhook URL

- OmniHSS에서 Webhook URL을 설정
- Webhook URL을 설정

Webhook Event

Webhook Event ID (event\_id)를 포함

```

const processedEvents = new Set();

app.post('/omnihss-webhook', (req, res) => {
 const eventId = req.body.event_id;

 if (processedEvents.has(eventId)) {
 // 重複
 return res.status(200).json({ status: 'duplicate' });
 }

 processedEvents.add(eventId);

 // 処理 webhook...
 processWebhook(req.body);

 res.status(200).json({ status: 'processed' });
});

```

## Webhook 問題

問題

- 発生 HTTP 4xx ~ 5xx
- HSS から webhook 失敗

原因

### 1. 401 Unauthorized - 認証失敗

- 正しい webhook URL ではない
- 署名が一致しない

### 2. 400 Bad Request - 不正なリクエスト

- 正しい webhook URL ではない
- 正しい Content-Type ではない

### 3. 500 Internal Server Error - サーバエラー

- サーバエラー

- 認證簽名

認證

認證簽名

```
app.post('/omnihss-webhook', async (req, res) => {
 try {
 // 認證
 if (!verifyWebhook(req)) {
 return res.status(401).json({ error: 'Invalid signature' });
 }

 // 認證
 if (!req.body.event || !req.body.subscriber) {
 return res.status(400).json({ error: 'Invalid payload' });
 }

 // 處理 webhook
 await processWebhook(req.body);

 res.status(200).json({ status: 'ok' });

 } catch (error) {
 console.error('Webhook processing error:', error);
 // 200 認證失敗
 res.status(200).json({ status: 'error', message: error.message });
 }
});
```

認證簽名

認證

- 處理 webhook 認證簽名
- 認證簽名 null 認證

認證

1. 認證簽名 - 認證簽名IMS認證

## 2. 接收数据 - 接收 webhook 数据

接收数据

接收数据

```
const { subscriber } = req.body;

// 接收数据
const imsProfile = subscriber.ims_profile || { name: 'No IMS' };
const roamingProfile = subscriber.roaming_profile || { name: 'No Roaming' };

// 接收 MSISDN
const msisdns = subscriber.msisdns || [];
```

接收数据

## Webhook 接收

接收 webhook 数据

接收数据

- Webhook 接收数据
- Webhook 接收数据
- 接收数据
- 接收数据
- 接收数据

接收数据 Prometheus/Grafana

```
Webhook 统计
rate(omnihss_webhook_success_total[5m]) /
rate(omnihss_webhook_attempts_total[5m])

Webhook 延迟
histogram_quantile(0.95, omnihss_webhook_duration_seconds)
```

# Webhook 配置

配置 webhook 接收来自设备的数据

配置项

```
{
 "timestamp": "2025-01-15T14:30:00Z",
 "level": "info",
 "component": "webhook",
 "event_id": "550e8400-e29b-41d4-a716-446655440000",
 "webhook_id": "4e5d6c7b-8a9f-4b0c-8d1e-2f3a4b5c6d7e",
 "event_type": "update_location_request",
 "subscriber_imsi": "001001123456789",
 "endpoint": "https://your-server.com/omnihss-webhook",
 "http_status": 200,
 "duration_ms": 145,
 "error": null
}
```



