← 概述

OmniUDM 配置

OmniUDM 在启动时通过 `config/runtime.exs` 从环境变量进行配置，位于 `:omniudm` 应用程序键下。每个值都有内置的默认值，因此该过程在没有设置环境的情况下开始；生产部署应至少设置 SBI、NRF、UDR、PLMN 和 `hnet_key_dir` 值。

示例配置

```
config :omniudm,
  udr_uri: "http://127.0.0.22:7777",
  sbi_scheme: "http",
  sbi_addr: "127.0.0.12",
  sbi_port: 7777,
  nrf_uri: "http://127.0.0.1:7777",
  mcc: "999",
  mnc: "70",
  prometheus_metrics_port: 9568,
  heartbeat_interval: 10_000,
  hnet_key_dir: "/etc/omniudm/hnet"

config :logger, :default_formatter,
  format: {OmniLogger.JsonFormatter, :format},
  metadata: :all
```

参数参考

参数	类型	必需	默认	
`udr_uri`	string (URI)	否	`http://127.0.0.22:7777`	`UDR_U`
`sbi_scheme`	string	否	`http`	`SBI_S`
`sbi_addr`	string (IP)	否	`127.0.0.12`	`SBI_A`
`sbi_port`	integer	否	`7777`	`SBI_P`
`nrf_uri`	string (URI)	否	`http://127.0.0.1:7777`	`NRF_U`
`mcc`	string	否	`999`	`MCC`

参数	类型	必需	默认	
`mnc`	string	否	`70`	`MNC`
`prometheus_metrics_port`	integer	否	`9568`	`PROM`
`heartbeat_interval`	integer (ms)	否	`10000`	`HEAR`
`hnet_key_dir`	string (path)	否	`/etc/omniudm/hnet`	`HNET_`

可选调优键

这些键从应用程序配置中读取，并未在 `runtime.exs` 中作为环境变量显示；如果部署需要覆盖默认值，请在配置中设置它们。

参数	类型	默认	描述
`default_nssai`	map	eMBB / URLLC / mIoT 实验切片（见下文）	当 UDR `am-data` 响应没有携带显式 `nssai` 时，返回的后备 NSSAI 订阅。设置此值以宣传网络的真实切片，包括 SD 合格的 S-NSSAIs
`nf_capacity`	integer (0-65535)	应用程序默认	此 NF 实例在与 NRF 注册的 NF 配置文件中宣传的相对处理能力。消费者根据能力加权 NF 选择。可通过 OAM 配置 PATCH 端点在运行时调整（`PATCH /api/oam/config/{id}`）；通常没有专用的环境变量
`nf_priority`	integer (0-65535)	应用程序默认	此 NF 实例在与 NRF 注册的 NF 配置文件中宣传的相对选择优先级（值越低优先级越高）。可通过 OAM 配置 PATCH 端点在运行时调整（`PATCH /api/oam/config/{id}`）；通常没有专用的环境变量

内置的 `default_nssai` 后备宣传 `defaultSingleNssais` 为 `sst 1`（和 `sst 1 / sd 000001`）以及覆盖 `sst 1`、`sst 1 / sd 000001`、`sst 2 / sd 000002`（URLLC）和 `sst 3 / sd 000003`（mIoT）的 `singleNssais`。如果提供给 AMF 的切片未配置为服务，则会以 5GMM 原因 #62 拒绝该用户，因此此后备必须与网络实际提供的服务相匹配。

该值是一个 `Nssai` 对象（TS 29.503 / TS 29.571），具有以下子字段：

字段	类型	语义
`defaultSingleNssais`	list of S-NSSAI	UE 被允许使用的切片 □□ 明确请求它们 - 当 UE 不请求 NSSAI 时，AMF 应用的订阅默认集。必须是 `singleNssais` 的子集
`singleNssais`	list of S-NSSAI	UE 在此 PLMN 上有权享有的完整订阅切片集

每个 S-NSSAI 条目是一个映射：

字段	类型	语义
`sst`	integer (0–255)	切片/服务类型。标准化值：`1` = eMBB，`2` = URLLC，`3` = mIoT
`sd`	string (6 hex digits)	可选切片区分符，限定 `sst`。对于仅 SST 的切片省略；网络仅以 SD 合格形式提供的 SST 必须列出其 `sd`（未合格的条目将不匹配）

在 `defaultSingleNssais` 中存在但在 `singleNssais` 中不存在的 S-NSSAI，或者此处的切片没有 AMF 配置为服务，将导致 AMF 以 5GMM 原因 #62 拒绝 UE。

家庭网络密钥（`hnet_key_dir`）

`hnet_key_dir` 是存放用于 ECIES Profile A/B SUCI 解密的家庭网络私钥的目录。每个 SIM 都配置有家庭网络公钥及其密钥标识符（`home_network_public_key_id`）；OmniUDM 必须在该相同标识符下持有匹配的私钥。

密钥文件在首次使用时加载一次并缓存到内存中���不存在的目录是可以容忍的（会记录警告，ECIES 解密实际上被禁用 - Null 方案 SUCI 仍然有效）。

文件命名。密钥标识符和曲线取自文件名。接受两种约定：

约定	**Profile A (Curve25519)**	**Profile B (P-256)**
点分 `<id>.<curve>.key`	`1.x25519.key`	`2.secp256r1.key`
连字符 `<curve>-<id>.key`	`curve25519-1.key` 或 `x25519-1.key`	`secp256r1-2.key`

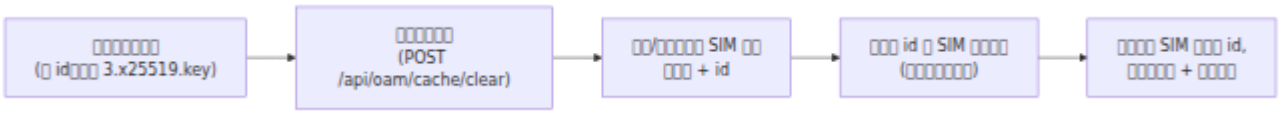
数字 `<id>` 必须与 SIM 上配置的 `home_network_public_key_id` 匹配。不匹配已识别模式的文件以及没有 `.key` 扩展名的文件将被跳过并记录警告。

文件内容。每个文件以三种编码之一保存原始私钥，自动检测：

- **PEM** - PKCS#8 `-----BEGIN PRIVATE KEY-----` SEC1 `-----BEGIN EC PRIVATE KEY-----`
- **Hex** -
- **Raw** -

SIM SUCI SIM SUPI `suci-0-<mcc>-<mnc>-<routing-ind>-<scheme>-<home_nw_pub_key_id>-<scheme_output>`OmniUDM SUCI `<home_nw_pub_key_id>` `<id>` - SIM - SUCI `unknown_home_network_key_id`

/



SIM MAC SIM SIM SIM

- `hnet_key_dir` - SUCI OmniUDM
- SIM MAC
- OAM `POST /api/oam/cache/clear`

/ OAM API

HTTPS /OAM API `8443` TLS SBI

方法	路径	目的
GET	`/api/status/api`	管理 API 状态
GET	`/api/status/license`	许可证状态
GET	`/api/status/nrf`	NRF 注册状态
GET	`/api/status/nf`	NF 状态摘要
GET	`/api/ue`	列出活动的内存中 UE 上下文
GET	`/api/ue/{supi}`	按 SUPI 检查单个 UE 上下文
GET	`/api/statistics`	服务统计信息，包括活动内存中 UE 上下文的实时计数
GET	`/api/oam/config`	读取当前有效配置
PATCH	`/api/oam/config/{id}`	更新配置值（PUT 也被接受）
DELETE	`/api/oam/ue_cache/{id}`	从内存缓存中驱逐 UE 上下文
POST	`/api/oam/ue/sdm-notify`	手动触发 SUPI 的 SDM 数据变更通知。向每个活动 SDM 订阅的 `callbackReference` POST 一个 `ModificationNotification`（例如，在带外配置更改后）
DELETE	`/api/oam/ue/amf-registration/{id}`	强制 UE 的 AMF 注销。清除存储的 AMF 3GPP 注册□通知服务 AMF 其 `deregCallbackUri` 以释放 UE（注销原因 `SUBSCRIPTION_WITHDRAWN`）；如果 UE 没有 AMF 注册，则返回 404
POST	`/api/oam/cache/clear`	清除缓存（包括家庭网络密钥缓存）
POST	`/api/oam/log_level`	更改运行时日志级别（`{"level": ...}`）

方法	路径	目的
GET	`/api/oam/resources`	资源清单
GET	`/api/oam/diagnostics`	诊断快照
POST	`/api/oam/nrf/reregister`	强制与 NRF 重新注册

日志记录

设置	值	描述
`format`	`{OmniLogger.JsonFormatter, :format}`	日志行作为结构化 JSON 通过共享的 OmniLogger 格式化程序发出，适合日志聚合器的摄取
`metadata`	`:all`	每行中包含所有日志元数据（包括每个请求会话上下文，如 SUPI 和过程）

运行时日志级别可以通过管理/OAM API 动态更改（`POST /api/oam/log_level`）。密钥材料（CK/IK/KAUSF/SQN⊕AK）永远不会被记录；仅非秘密上下文如服务网络名称和解密的 SUPI 出现在日志中。

← 概述

OmniUDM 指标

指标以 Prometheus 格式在 `prometheus_metrics_port`（默认 `9568`）的 `/metrics` 上公开。

UDM 服务指标

指标	类型	标签	
`omni_udm_auth_requests_count`	counter	`supi`, `result`	按结果（`suc` `fail` 证向量[
`omni_udm_auth_vector_generations_total`	counter	`result`	按结果[试
`omni_udm_auth_result_count`	counter	`supi`, `result`	通过 `a` `even` 证结果[（`res` 的 `su` 值）
`omni_udm_uecm_registrations_total`	counter	`access_type`	跨两种[UEC 册。3C 3GPI 路径都[按 `acce` （`3gp` `non_` 记
`omni_udm_sdm_queries_total`	counter	`data_type`	按数据[数据查[`data` `data` `sele` `data` `ue-c`

指标	类型	标签	
			`in-s` `data` `sets`
`omni_udm_active_ue_contexts_count`	gauge	none	活跃的 下文数 下文创
`omni_udm_nrf_registration_status`	gauge	`nf_type`	NRF = 已 注册）

有用的查询

```
# 5 钟内的认证失败率
sum(rate(omni_udm_auth_requests_count{result="failure"}[5m]))
  / sum(rate(omni_udm_auth_requests_count[5m]))

# 按数据类型的 SDM 查询率
sum by (data_type) (rate(omni_udm_sdm_queries_total[5m]))

# 按接入类型（3gpp / non_3gpp）拆 的 UECM 注册率
sum by (access_type) (rate(omni_udm_uecm_registrations_total[5m]))

# 认证向量生成失败率
sum(rate(omni_udm_auth_vector_generations_total{result="failure"}
[5m]))
  / sum(rate(omni_udm_auth_vector_generations_total[5m]))

# 如果 UDM 未注册到 NRF，则发出警报
min(omni_udm_nrf_registration_status) < 1
```

`omni_udm_active_ue_contexts_count` 跟踪活跃的内存中 UE 上下文的实时数量；在每次 UE 上下文创建和删除时更新。OAM 的 `/api/statistics` 和 `/api/status/nf` 端点按需报告相同的计数。

BEAM VM 指标

指标	类型	描述
`beam_memory_total`	gauge	总 BEAM 内存（字节）
`beam_memory_processes`	gauge	分配给 Erlang 进程的内存
`beam_memory_processes_used`	gauge	进程实际使用的内存
`beam_memory_system`	gauge	系统内存（ETS、原子、代码）
`beam_memory_atom`	gauge	总原子内存
`beam_memory_atom_used`	gauge	使用的原子内存
`beam_memory_binary`	gauge	二进制内存
`beam_memory_code`	gauge	代码内存
`beam_memory_ets`	gauge	ETS 表内存
`beam_processes_count`	gauge	Erlang 进程数量
`beam_ports_count`	gauge	Erlang 端口数量
`beam_atom_count`	gauge	原子数量
`beam_vm_uptime`	gauge	VM 运行时间（秒）

← 概述

OmniUDM 操作

OmniUDM 是 Omnitouch 5G 核心的统一数据管理 (UDM) 功能。它通过标准的 Nudm 服务向核心的其余部分提供用户身份、订阅和上下文数据，并将所有权威数据通过 `nudr-dr` 备份到统一数据存储库 (OmniUDR)。

3GPP 角色和规范参考

方面	参考
5GC 中的 UDM 功能角色	TS 23.501
Nudm 服务 (SDM / UEAU / UECM)	TS 29.503
Nudr_DataRepository (消费)	TS 29.504
SUCI / SUPI 隐藏和保护方案	TS 33.501 §6.12 和附录 C
SUCI / SUPI 结构和编码	TS 23.003 §2.2B
5G-AKA 认证和密钥层次	TS 33.501 §6.1.3, 附录 A
移动身份 (SUCI/MSIN BCD) 编码	TS 24.501

接口和 SBI 端点

OmniUDM 向 NRF 注册以下服务，并在 SBI 监听器 (`sbi_scheme://sbi_addr:sbi_port`) 上提供它们：

服务	版本	目的
`nudm-sdm`	v2	用户数据管理
`nudm-ueau`	v1	UE 认证向量生成
`nudm-uecm`	v1	UE 上下文 (AMF/SMF 注册) 管理

所有权威的用户和上下文数据通过 `nudr-dr/v2` 从 OmniUDR 读取/写入 (基本路径 `.../nudr-dr/v2/subscription-data/{ueId}/...`)。所有请求/响应体均为 JSON。错误以 `application/problem+json` ProblemDetails 的形式返回。

Nudm_UEAuthentication (`nudm-ueau/v1`)

方法	路径	描述
POST	`/{supiOrSuci}/security-information/generate-auth-data`	生成 5G-AKA 认证向量。接受 SUPI 或 SUCI; 首先解密 SUCI。需要 `servingNetworkName` 和 `ausfInstanceId`
POST	`/{supi}/auth-events`	记录认证结果事件。需要 `nfInstanceId`、`success`、`timeStamp`、`authType`、`servingNetworkName`
PUT	`/{supi}/auth-events/{authEventId}`	更新/删除认证事件 (AuthEvent 主体, `authRemovalInd=true` 清除它)

Nudm_UECM (`nudm-uecm/v1`)

方法	路径	描述
PUT	`/{supi}/registrations/amf-3gpp-access`	AMF 为 UE 注册 3GPP 访问
PATCH	`/{supi}/registrations/amf-3gpp-access`	更新存储的 AMF 3GPP 注册参数 (合并补丁)
GET	`/{supi}/registrations/amf-3gpp-access`	检索存储的 AMF 3GPP 注册
PUT	`/{supi}/registrations/amf-non-3gpp-access`	AMF 为 UE 注册非 3GPP 访问
GET	`/{supi}/registrations/amf-non-3gpp-access`	检索��储的 AMF 非 3GPP 注册
PUT	`/{supi}/registrations/smf-registrations/{pduSessionId}`	SMF 注册 PDU 会话上下文
DELETE	`/{supi}/registrations/smf-registrations/{pduSessionId}`	SMF 注销 PDU 会话上下文

Nudm_SDM (`nudm-sdm/v2`)

方法	路径	描述
GET	`/{supi}`	GetDataSets - 检索由强制 `dataset-names` 查询参数选择的多个数据集（以逗号分隔的 `AM`、`SM`、`SMF_SEL`、`UEC_SMF`）
GET	`/{supi}/am-data`	接入和移动性订阅数据（可选 `plmn-id` 查询）
GET	`/{supi}/sm-data`	会话管理订阅数据（可选 `plmn-id` 查询）
GET	`/{supi}/smf-select-data`	SMF 选择订阅数据（可选 `plmn-id` 查询）
GET	`/{supi}/nssai`	网络切片选择订阅数据
GET	`/{supi}/ue-context-in-smf-data`	UE 在 SMF 中的上下文（活动 PDU 会话）：见下文说明
POST	`/{supi}/sdm-subscriptions`	订阅数据变更通知。需要 `nfInstanceId`、`callbackReference`、`monitoredResourceUris`
GET	`/{supi}/sdm-subscriptions/{subscriptionId}`	检索存储的 SDM 订阅
PATCH	`/{supi}/sdm-subscriptions/{subscriptionId}`	修改 SDM 订阅（合并补丁）
DELETE	`/{supi}/sdm-subscriptions/{subscriptionId}`	移除 SDM 订阅

SUCI / SUPI 隐藏

为了保护用户的永久身份，UE 发送 **SUCI**（订阅隐藏标识符）而不是 **SUPI**（订阅永久标识符，即 `imsi-<IMSI>`）。当 AUSF 请求 OmniUDM 为 SUCI 生成认证向量时，OmniUDM 在查找 **UDR** 中的用户数据之前解密 **SUCI** 为 **SUPI**。

OmniUDM 支持 TS 33.501 §6.12 / 附录 C 中定义的三种保护方案：

方案	名称	曲线 / 加密	需要 **HNET** 密钥?
0	空方案	无 - MSIN 为明文	否
1	ECIES A 型	Curve25519 (X25519) + AES-128-CTR + HMAC-SHA-256	是
2	ECIES B 型	secp256r1 / P-256 + AES-128-CTR + HMAC-SHA-256	是

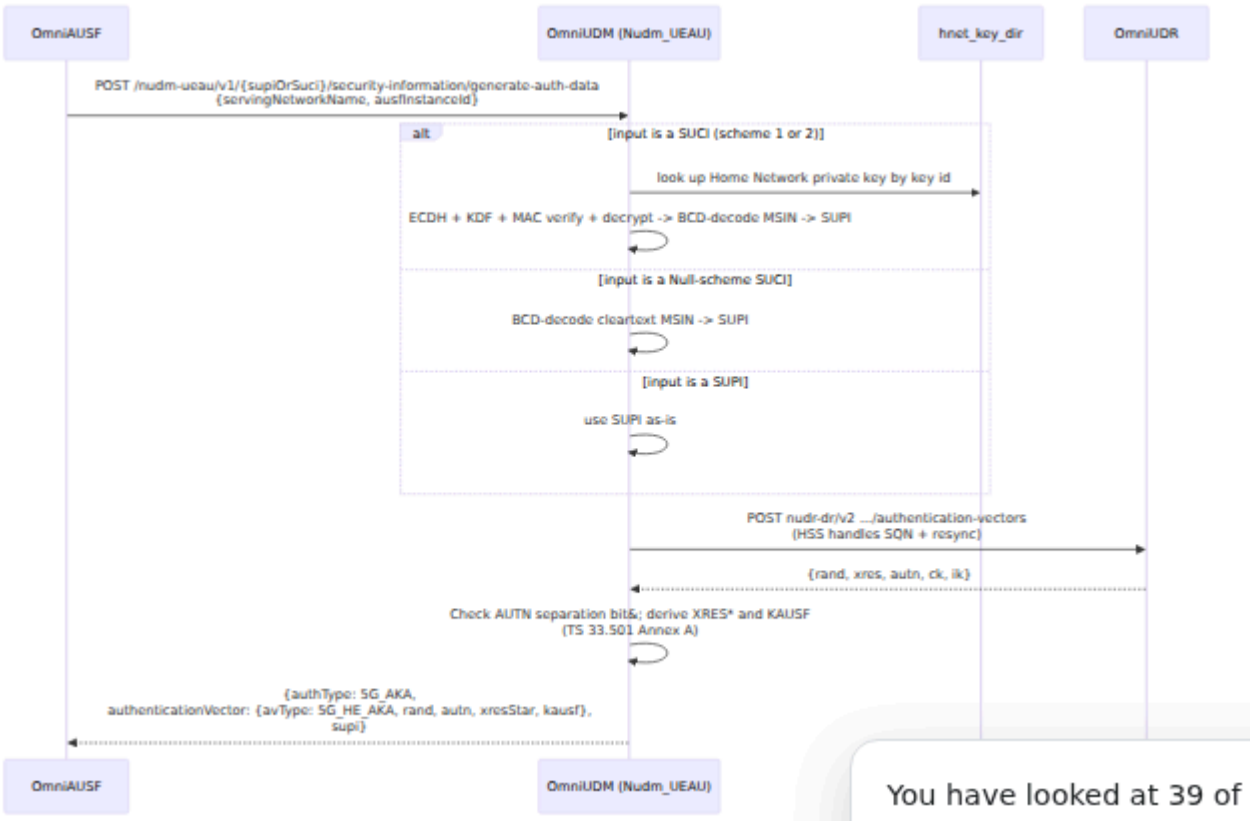
对于 空方案，MSIN 以明文形式传输；它从方案输出中 **BCD** 解码 以重构 IMSI（根据 TS 24.501 - MSIN 数字是 BCD 编码，而不是十六进制）。对于 **A** 型和 **B** 型，解密执行针对运营商的 家庭网络私钥 的短期-静态 ECDH，推导加密/MAC 密钥 (ANSI-X9.63-KDF 与 SHA-256)，验证 MAC，解密隐藏的 MSIN，并 BCD 解码以重构 IMSI。

A/B 型的家庭网络私钥从 `hnet_key_dir` 目录加载。文件命名、编码以及安全和缓存刷新要求在 配置 → 家庭网络密钥 (`hnet_key_dir`) 中进行了记录。

关键程序

SUCI 解密和认证向量生成

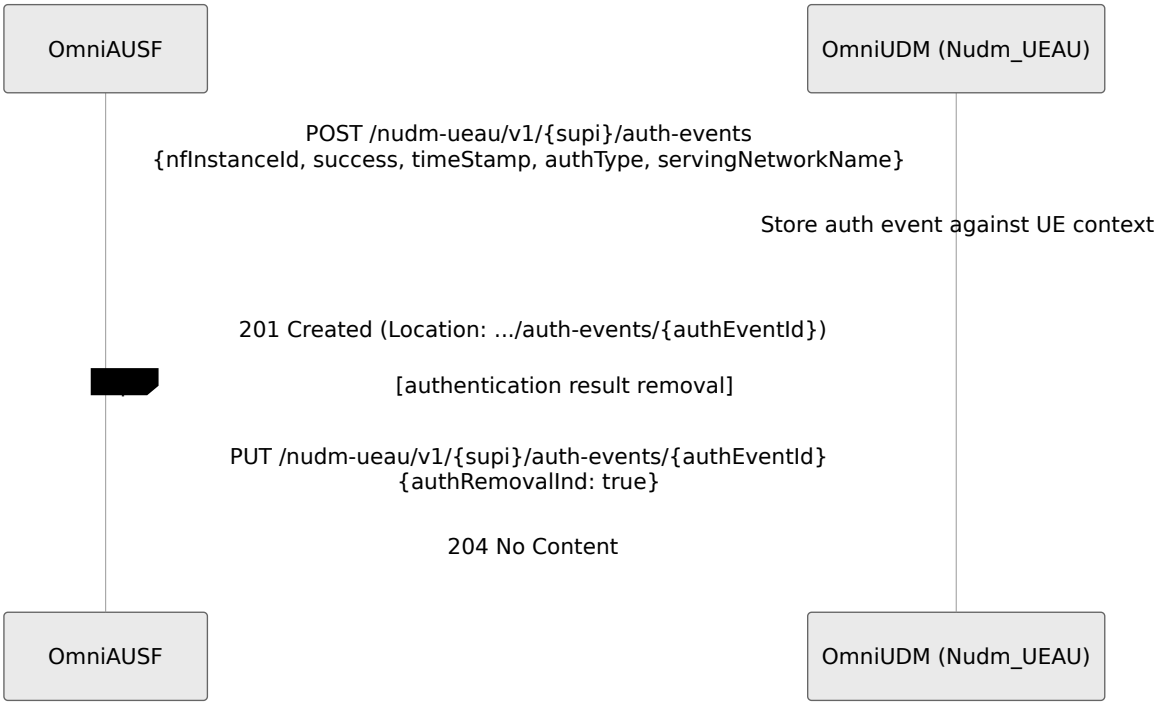
AUSF 请求认证向量；OmniUDM 解析身份 (��有必要，解密 SUCI)，通过 UDR 向 HSS 请求向量，然后从中推导 5G 密钥。**Ki/OPc**、**SQN** 管理和任何重新同步保持在 **HSS** 后面 **UDR** - OmniUDM 检查 HSS AUTN 是否携带 5G 分离位 (AMF 位 0，TS 33.501 §6.1.3.2，取自用户的预配置 AMF)，并从返回的 `xres`、`ck` 和 `ik` 中推导 `XRES*` 和 `KAUSF`。AUSF 提供的 `resynchronizationInfo` 通过 UDR 转发给 HSS。

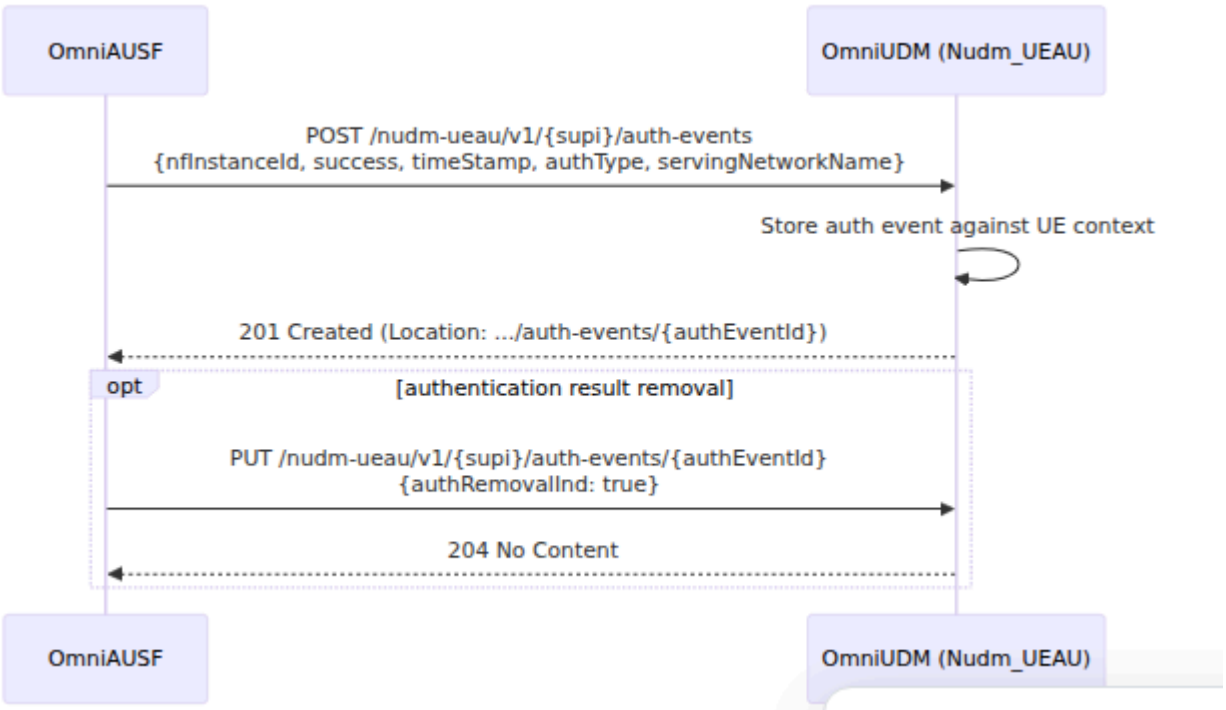


解密失败返回 HTTP 403（`SUCI 解密失败`），不支持的保护方案返回 HTTP 501，缺失或禁用的用户返回 HTTP 404（`USER_NOT_FOUND`），无效的 HSS 向量（包括没有分离位的 AUTN）或 HSS 向量生成失败返回 HTTP 500。

认证结果事件记录

在 AUSF 完成认证后，它记录结果。OmniUDM 将事件存储在 UE 上下文中，并返回创建事件的 `Location`；AUSF 可以稍后 `PUT` 一个 `AuthEvent`，并设置 `authRemovalInd=true` 来清除它（204 无内容）。

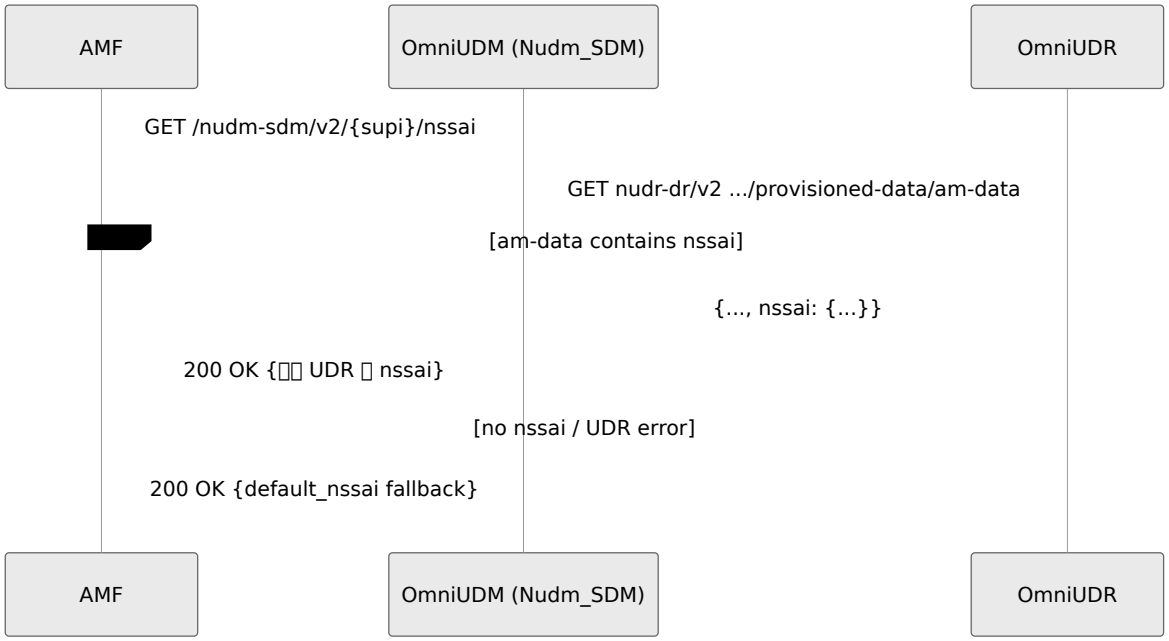




SDM 用户数据获取（`sm-data`）

注册后，AMF/SMF 获取订阅数据。OmniUDM 将请求代理到 UDR，添加服务 PLMN，并返回 UDR 响应。对于 `sm-data`，单个 UDR 对象根据 TS 29.503 `SmSubsData` 包装在数组中。

`am-data`、`smf-select-data` 和 GetDataSets（`GET /{supi}`）流程遵循相同的代理模式针对 UDR。GetDataSets 仅组装它可以解析的请求数据集，跳过任何出错的数据集。`nssai` 从 UDR `am-data` 响应中推导（见下文）。



> **UE-context-in-SMF** 返回实时 **PDU** 会话上下文。`GET /{supi}/ue-context-in-smf-data` 和 GetDataSets 中的 `UEC_SMF` 数据集返回 SMF 通过 Nudm_UECM SMF 注册程序报告的 UE 的活动 PDU 会话上下文。每个存储的 SMF 注册根据 TS 29.503 投影到 `pduSessions` 映射中（按 PDU 会话 ID 键入）。当 UE 确实没有 SMF 注册时，映射为空（`{"pduSessions": {}}`）。

NSSAI 获取与默认回退

`GET /{supi}/nssai` 从 UDR 获取 `am-data` 并提取嵌入的 `nssai`。如果 UDR 返回 `am-data` 而没有 `nssai`，或者 UDR 查找失败，OmniUDM 返回可配置的 `default_nssai` 回退，以便 AMF 始终接收切片列表。

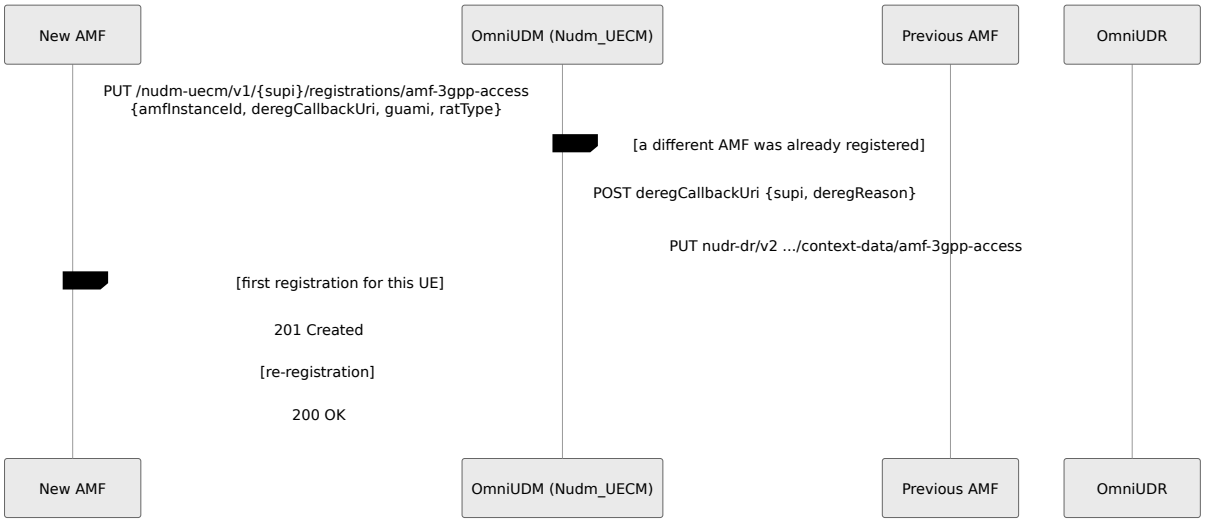
`default_nssai` 回退必须与网络实际提供的切片匹配 - 提供给 AMF 但未配置的切片的用户将被拒绝，原因是 5GMM 原因 #62。见 配置 → 可选调优键。

SDM 变更通知

具有活动 SDM 订阅的用户在后续变更时会收到通知：OmniUDM 在监控资源发生变化时异步 POST 一个 `ModificationNotification` 到每个订阅的 `callbackReference`。也可以通过管理/OAM API 手动触发通知。

UECM AMF 注册和移动性

当 UE 注册时，AMF 会在 OmniUDM 中为该 UE 注册。如果之前注册了不同的 AMF，OmniUDM 会通知旧 AMF 的 `deregCallbackUri` 以便其释放 UE，然后将新注册持久化到 UDR。



强制信息元素根据 TS 29.503 进行验证：AMF 3GPP 注册需要 `amfInstanceId`、`deregCallbackUri`、`guami`、`ratType`；SMF 注册需要 `smfInstanceId`、`pduSessionId`、`singleNssai`、`plmnId`，并且主体 `pduSessionId` 必须与路径匹配（不匹配返回 `MANDATORY_IE_INCORRECT`）。缺失的信息元素返回 HTTP 400。

← 概述

OmniUDM 故障排除

身份验证失败，404 用户未找到

OmniUDM 已解析身份，但 UDR 未返回身份验证订阅。确认 `udr_uri` 可访问，OmniUDR 可以访问订阅者数据库，并且 IMSI 已配置。对于 SUCI 输入，确认去隐私化产生了预期的 IMSI（检查“去隐私化 SUCI 到...”日志行）。

身份验证失败，403 SUCI 去隐私化失败

OmniUDM 无法去隐私化 SUCI。详细字段说明了具体原因：

- `unknown_home_network_key_id` - 没有加载 SUCI 的密钥标识符的私钥。确认 `hnet_key_dir` 中存在该 `home_network_public_key_id` 的密钥文件，并遵循公认的命名约定，并且自从添加文件以来密钥缓存已被刷新。
- `ecies_mac_verification_failed` - 加载的私钥与 SIM 上的公钥不匹配（密钥错误或交换），或 SUCI 已损坏。验证 SIM 上配置的密钥对与该标识符的部署私钥是否一致。
- `ecies_profile_a_scheme_output_too_short` / `ecies_profile_b_scheme_output_too_short` - SUCI 方案输出被截断 / 格式错误。
- `invalid_suci_format` - SUCI 字符串格式不正确。

身份验证失[illegible]，501 不支持的保护方案

SUCI 使用了除 0（空）、1（方案 A）或 2（方案 B）之外的保护方案。仅支持这三种。

身份验证失败，500 内部错误

HSS 向量字段（`rand`/`xres`/`ck`/`ik`/`autn`）缺失、无效十六进制或长度错误；AUTN 缺少 5G 分离位；或 HSS 向量生成失败。日志行 `Invalid auth key material ... separation_bit_not_set` 表示订阅者在 HSS 密钥集中配置的 AMF 的最高位未设置（使用 `8000`）。否则，请验证 HSS 中的订阅者密钥集，并确保 OmniUDR 可以访问 HSS。

`am-data` / `sm-data` / `smf-select-data` 返回 404 未找到

UDR 查找该数据集失败。确认订阅者在服务 PLMN 中配置的数据存在于 UDR 中，`mcc`/`mnc`（或请求的 `plmn-id`）与数据的配置方式匹配，并且 `udr_uri` 可访问。

UE 被 AMF 拒绝，5GMM 原因 #62（没有支持的切片）

提供给 AMF 的 NSSAI 不包括 AMF 配置为服务的切片。如果订阅者的 UDR `am-data` 不携带 `nssai`，OmniUDM 返回 `default_nssai` 备用 - 设置 `default_nssai`（或在 UDR 中配置 `nssai`）以匹配网络的真实切片，包括任何 SD 合格的 S-NSSAIs。

重启后 UE 上下文或注册不存在

每个 UE 的运行时状态（身份验证上下文、AMF/SMF 注册、SDM 订阅）保存在��存上下文存储中，并通过重新注册重建；权威的订阅者数据保留在 OmniUDR 中。重启后，AMF/SMF 在下一个定期注册更新、PDU 会话重新建立或 UE 断电周期时重新注册。如果特定注册比预期更早缺失，请触发消费者 NF 的注册更新，而不是等待定期计时器。

未收到 SDM 数据变更通知

确认订阅的 `callbackReference` URI 可以从 OmniUDM 主机访问。通知是异步发送的；查找“SDM 通知到 … 失败”的警告。可以通过管理/OAM API 手动触发 SUPI 的通知（`POST /api/oam/ue/sdm-notify`）。

更改家庭网络密钥后，去隐私化仍使用旧密钥

家庭网络密钥在首次加载后会缓存到内存中。在 `hnet_key_dir` 中添加或替换密钥文件后，通过 OAM 管理操作热重载它们（`POST /api/oam/hnet/reload`）；响应报告加载的密钥数量及其密钥 ID。无需重启进程。

UDM 无法被其他 NF 发现

检查 `omni_udm_nrf_registration_status`。如果为 `0`，请确认 `nrf_uri` 可访问且心跳成功；OAM 管理 API 提供 NRF 重新注册操作（`POST /api/oam/nrf/reregister`）。