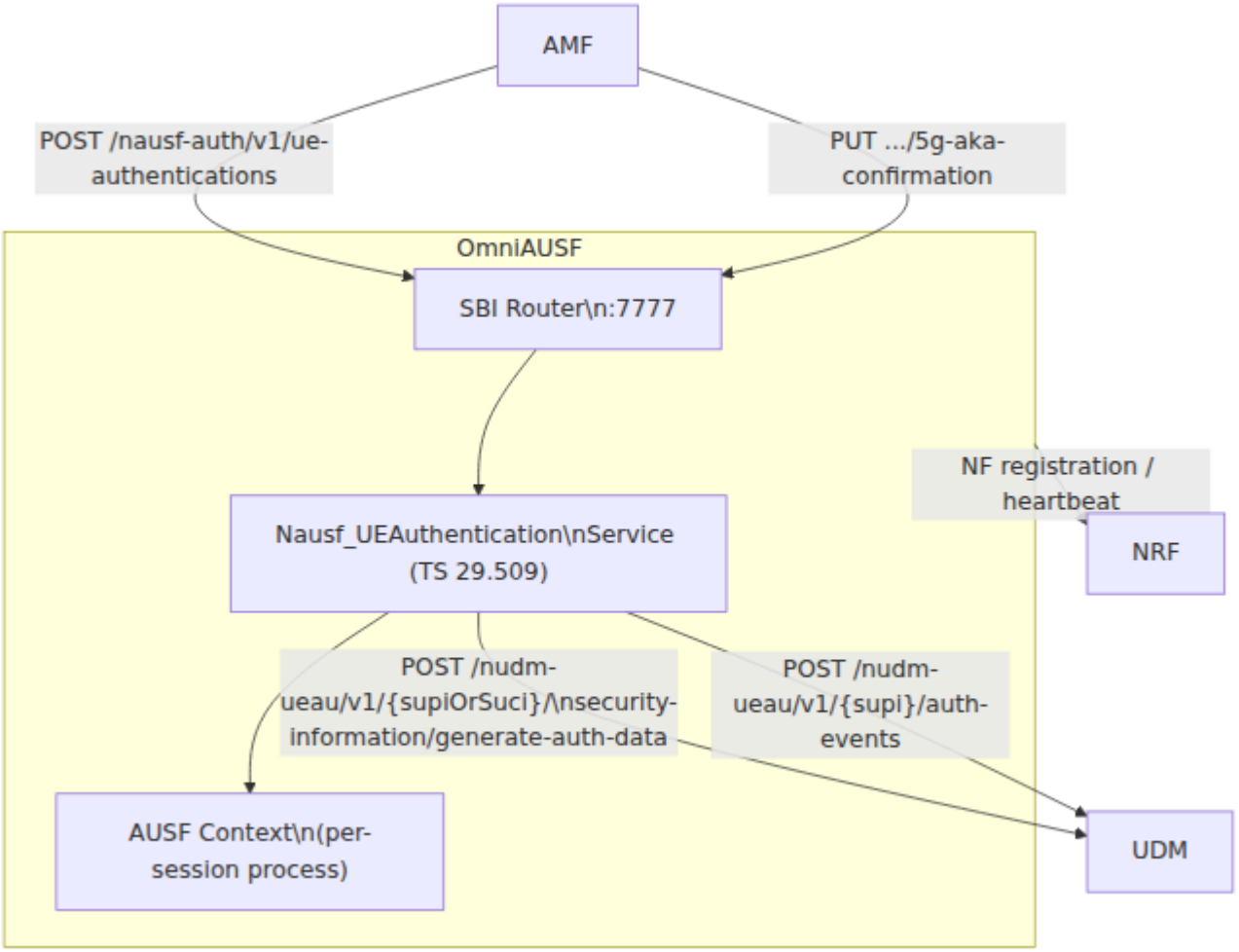
Opérations OmniAUSF

1. Vue d'ensemble des composants

OmniAUSF est la Fonction de Serveur d'Authentification (AUSF) autonome pour le cœur 5G d'Omnitouch. Il orchestre l'authentification 5G-AKA entre l'AMF et l'UDM, vérifiant les réponses d'authentification de l'UE et dérivant des clés de session. OmniAUSF était auparavant co-localisé au sein d'OmniUDM et est maintenant déployé en tant que NF indépendant avec son propre point de terminaison SBI.

Chaque session d'authentification est gérée par un processus dédié (processus-par-session-auth). Le contexte d'authentification est conservé en mémoire pendant la durée de l'échange d'authentification et supprimé à la fin ou en cas d'échec.


AMF
POST /nausf-auth/v1/ue-authentications
PUT .../5g-aka-confirmation
OmniAUSF
SBI Router\n:7777
NF registration / heartbeat
NRF
Nausf_UEAuthentication\nService (TS 29.509)
POST /nudm-ueau/v1/{supiOrSuci}/\nsecurity-information/generate-auth-data
POST /nudm-ueau/v1/{supi}/auth-events
AUSF Context\n(per-session process)
UDM

2. Rôle 3GPP et références de spécifications

Aspect	Référence
Définition fonctionnelle de l'AUSF	TS 23.501 Section 6.2.8
Service Nausf_UEAuthentication	TS 29.509
Authentification 5G-AKA	TS 33.501 Section 6.1.3
Calcul HXRES*/HRES*	TS 33.501 Annexe A.5
Dérivation KSEAF	TS 33.501 Annexe A.6
Génération de données d'authentification UDM	TS 29.503 Section 5.2.2
Resynchronisation SQN	TS 33.102 Section 6.3.5, TS 33.501 Section 6.1.3.4

3. Points de terminaison SBI

Tous les points de terminaison sont HTTP/1.1 avec `Content-Type: application/json`.

Nausf_UEAuthentication (TS 29.509)

Méthode	Chemin	Description	Succès
POST	`/nausf-auth/v1/ue-authentications`	Initier l'authentification de l'UE (AMF -> AUSF)	201 Créé
PUT	`/nausf-auth/v1/ue-authentications/{authCtxId}/5g-aka-confirmation`	Confirmer 5G-AKA (l'AMF envoie RES*)	200 Ok

Réponses d'erreur

Statut HTTP	Cause	Condition
404	USER_NOT_FOUND	L'UDM a retourné 404 pour l'abonné
401	AUTHENTICATION_FAILURE	HRES* ne correspond pas à HXRES*
500	SYSTEM_FAILURE	Erreur interne ou UDM inaccessible

4. Référence de configuration

OmniAUSF est configuré via l'environnement d'application Elixir sous la clé `:omniausf`.

Exemple de configuration

```
config :omniausf,
  sbi_scheme: "http",
  sbi_addr: "127.0.0.19",
  sbi_port: 7777,
  nrf_uri: "http://127.0.0.10:7777",
  udm_uri: "http://127.0.0.12:7777",
  mcc: "999",
  mnc: "70",
  heartbeat_interval: 10_000
```

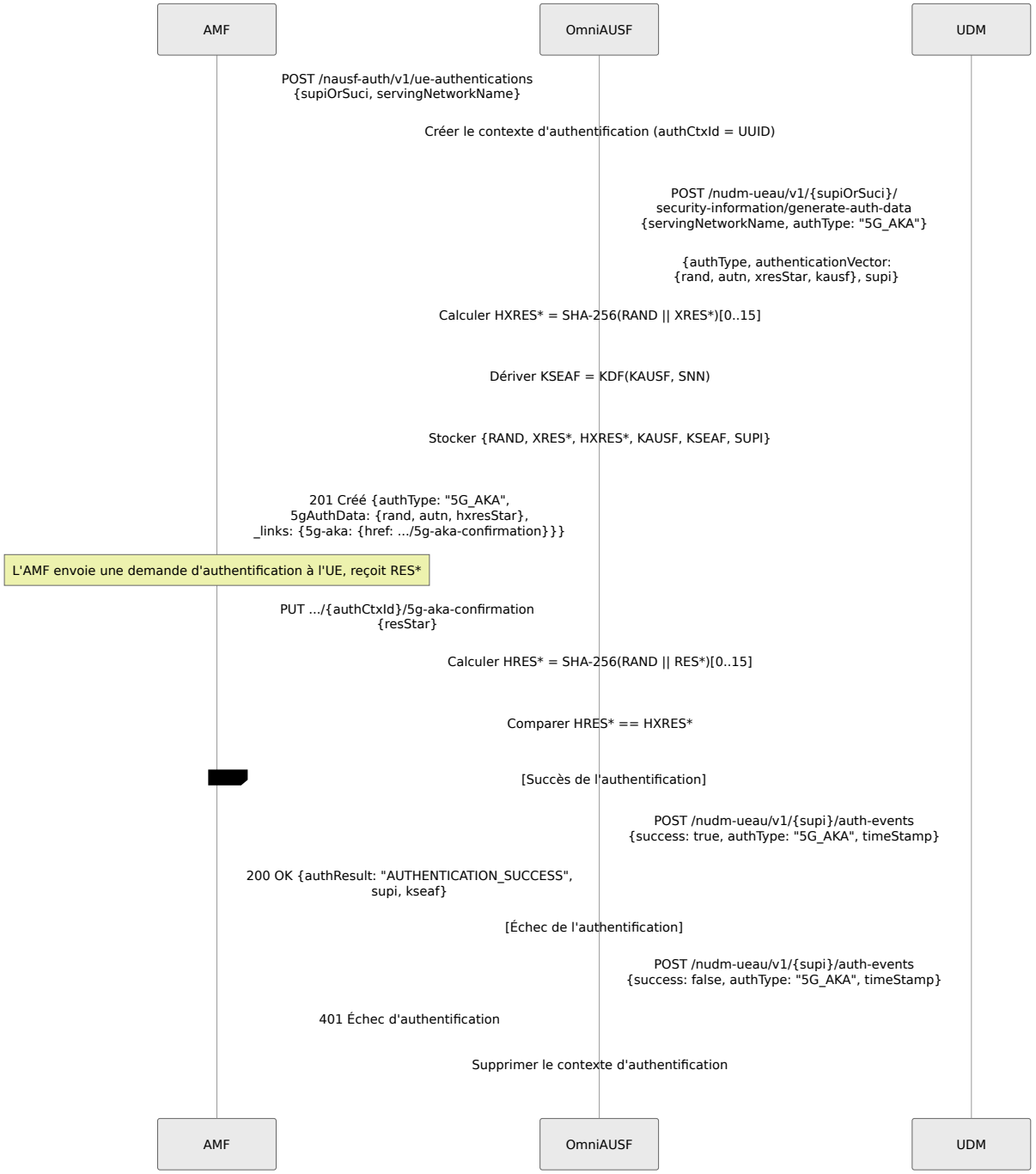
Tableau des paramètres

Paramètre	Type	Par défaut	Descriptio
`sbi_scheme`	chaîne	`"http"`	Schéma URI p le serveur HT SBI
`sbi_addr`	chaîne	`"127.0.0.19"`	Adresse IP à laquelle le serveur HTTP est lié
`sbi_port`	entier	`7777`	Port TCP sur lequel le serv HTTP SBI écou
`nrf_uri`	chaîne	`"http://127.0.0.10:7777"`	URI de base d NRF pour l'enregistrem NF et le heartbeat
`udm_uri`	chaîne	`"http://127.0.0.12:7777"`	URI de base d l'UDM pour la génération de vecteurs d'authentifica et le stockage d'événements d'authentifica
`mcc`	chaîne	`"999"`	Code de pays mobile pour l PLMN de serv

Paramètre	Type	Par défaut	Descriptio
`mnc`	chaîne	`"70"`	Code de rése mobile pour l PLMN de serv
`heartbeat_interval`	entier (ms)	`10000`	Intervalle auc OmniAUSF envoie des requêtes PAT de heartbeat NRF

5. Procédures clés

5.1 Flux d'authentification 5G-AKA



6. Métriques Prometheus

Métriques AUSF

Métrique	Type	Tags	Descri
`omni_ausf.auth.count`	compteur	`result`	Total des opé d'authentific (initiées/succ
`omni_ausf.nrf.registration.status`	jauge	`nf_type`	Statut d'enre NRF (1=regis 0=non)
`omni_ausf.active_contexts.count`	jauge	--	Nombre de c d'authentific AUSF actifs

Métriques BEAM VM

Métrique	Type	Description
`beam.memory.total`	jauge	Mémoire totale BEAM en octets
`beam.memory.processes`	jauge	Mémoire utilisée par les processus Erlang
`beam.memory.system`	jauge	Mémoire système (ETS, atomes, code)
`beam.processes.count`	jauge	Nombre de processus Erlang
`beam.vm.uptime`	jauge	Temps de fonctionnement de la VM en secondes

7. Limitations connues

ID	Domaine	Description
AUSF-1	État en mémoire	Les contextes d'authentification sont stockés uniquement en mémoire. L'état est perdu lors du redémarrage du processus. Les sessions d'authentification actives échoueront lors du redémarrage de l'AUSF ; l'AMF doit réinitialiser l'authentification
AUSF-2	EAP-AKA'	Seule l'authentification 5G-AKA est prise en charge. La méthode d'authentification EAP-AKA' (TS 33.501 Section 6.1.3.1) n'est pas implémentée
AUSF-3	Transfert de resync	L'AUSF ne gère pas indépendamment `resynchronizationInfo` ; il le transmet à l'UDM. L'AMF doit inclure `resynchronizationInfo` dans la demande d'authentification initiale

8. Dépannage

L'authentification échoue avec 404 Utilisateur non trouvé

L'UDM a retourné 404 pour l'abonné. Confirmez :

1. `udm_uri` est accessible depuis l'hôte OmniAUSF.
2. L'IMSI de l'abonné existe dans le backend UDM/UDR/HSS.
3. Le SUCI présenté par l'AMF est correctement formaté.

L'authentification échoue avec 401 Échec d'authentification

L'AUSF a calculé HRES* à partir du RES* reçu et il ne correspondait pas au HXRES* stocké. Cela indique que les identifiants de l'UE (Ki, OPc) ne correspondent pas à ceux du backend, ou que le RAND/AUTN a été corrompu en transit.

UDM inaccessible (500 Erreur interne)

Vérifiez la configuration de `udm_uri` et la connectivité réseau. L'AUSF journalise `AUSF auth failed for {supiOrSuci}: {reason}` en cas d'échec de communication avec l'UDM.

Contexte d'authentification non trouvé lors de la confirmation

L'`authCtxId` dans la requête PUT ne correspond à aucun contexte actif. Les contextes sont supprimés après une confirmation réussie ou échouée, et sont perdus lors du redémarrage de l'AUSF. L'AMF doit réinitialiser l'authentification.