27/10/2025

DH-CNS

Réunion technique industriels

CNS
d'Gesondheetskeess

Mentions légales

Informations de nature générale

Les informations reprises dans cette présentation sont de nature générale. Elles ne sont pas adaptées à des circonstances personnelles ou spécifiques. Elles ne peuvent pas être considérées comme des conseils personnels, professionnels ou juridiques. Si vous avez besoin de conseils personnels ou spécifiques, consultez un des services de la CNS ou de vos caisses de maladie du secteur public, en fonction du thème de votre demande.

Informations officielles

Seuls les textes officiels publiés au Mémorial sont considérés comme authentiques. Les informations et les documents disponibles au sein de cette présentation ne peuvent pas être considérés comme une reproduction authentique des textes officiels. En cas de différences, le texte officiel publié au Mémorial prime toujours.

Contenu et responsabilité

La CNS consent de gros efforts pour que les informations mises à disposition au sein de cette présentation soient complètes et correctes. Ces informations n'ont cependant pas une vocation d'exhaustivité ou de constituer un engagement de la part de la CNS. Si vous désirez poser des questions au sujet de cette présentation, vous pouvez nous contacter à cns@secu.lu.

Informations d'autres autorités, instances et organisations

Cette présentation peut comporter des informations d'autres autorités, instances et organisations sur lesquels la CNS n’exerce aucun contrôle. La CNS n’offre aucune garantie quant au caractère exhaustif ou exact du contenu de ces informations. La CNS décline toute responsabilité pour les dommages directs ou indirects résultant de l'utilisation des ces informations et de leurs contenus.

Sommaire

- Introduction
- Authentification OIDC AeS
- eConnector SaaS
- Roadmap
- Questions/réponses

Introduction

Introduction

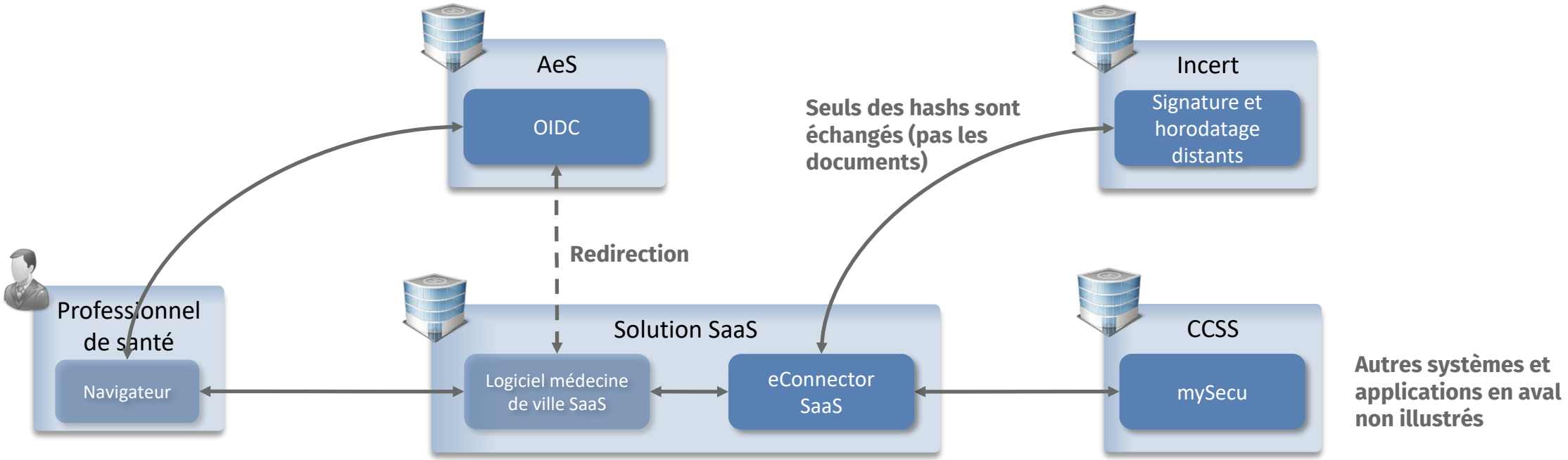
- Nouveautés eConnector
  - Utilisation de l'authentification OIDC de l'AeS.
  - Introduction de la signature distante en lieu et place des keystores locaux.
  - Support complet des logiciels de médecine de ville fonctionnant en mode SaaS.
  - Génération des PDFs directement dans l'eConnector via des templates CNS prédéfinis.

eConnector Desktop VS eConnector SaaS (1/2)

- eConnector Desktop
  - Est prévu pour fonctionner avec des logiciels de médecine de ville qui s'exécutent sur le poste du professionnel de santé
  - S'exécute directement sur le poste du professionnel de santé
- eConnector SaaS
  - Est prévu pour fonctionner avec des logiciels de médecine de ville qui s'exécutent en mode SaaS
    - Le logiciel de médecine de ville ne s'exécute donc **pas** sur le poste du professionnel de santé.
    - Ce dernier interagit avec le logiciel de médecine de ville à l'aide d'un navigateur qui communique avec le logiciel de médecine qui s'exécute à distance sur un **serveur**.
  - S'exécute sur un **serveur de l'éditeur** entre le logiciel de médecine de ville d'une part, et le service de signature à distance et mySecu d'autre part.

eConnector Desktop VS eConnector SaaS (2/2)

- Exemple de déploiement



Authentification OIDC AeS

Contexte

- eConnector Desktop
  - Authentification OIDC embarquée dans l'eConnector
  - ➔ slides suivants **non** applicables
- eConnector SaaS
  - Authentification OIDC doit être faite par le logiciel de médecine de ville
  - ➔ slides suivants applicables

Prérequis

- Besoin d'un client OIDC (client_id, client_secret)
- => Adresser demande à support_labelisation@incert.lu - en utilisant procédure communiquée
  - Demande doit spécifier une URL de redirection (redirect_uri) vers laquelle le serveur d'authentification de l'AeS redirigera l'utilisateur après authentification
  - Elle permet d'éviter les redirections malveillantes.
  - Cela **doit** être une URL à laquelle votre logiciel est accessible et dont le nom de domaine vous appartient.
  - **Exemple** : https://app.client.com/callback

1. Requête d'authentification (Authorization Code + PKCE)

Initialisation côté client (logiciel de médecine de ville)

L'éditeur redirige le PS vers le portail OIDC de l'AeS avec une requête contenant les informations suivantes :

- client_id : identifiant du client OIDC propre à l'éditeur (voir slide précédent)
- redirect_uri : URL de redirection du client OIDC (doit correspondre à celle indiquée dans la demande du client OIDC formulée par l'éditeur à l'AeS, voir slide précédent)
- scope : openid (valeur fixe)
- response_type : code (valeur fixe)
- code_challenge : valeur dérivée du code verifier
  - Code verifier est une valeur générée aléatoirement, voir https://www.rfc-editor.org/rfc/rfc7636#section-4.1
  - Dérivation du code_challenge, voir https://www.rfc-editor.org/rfc/rfc7636#section-4.2 (suivre la transformation S256)
- code_challenge_method : S256 (valeur fixe)

Exemple :

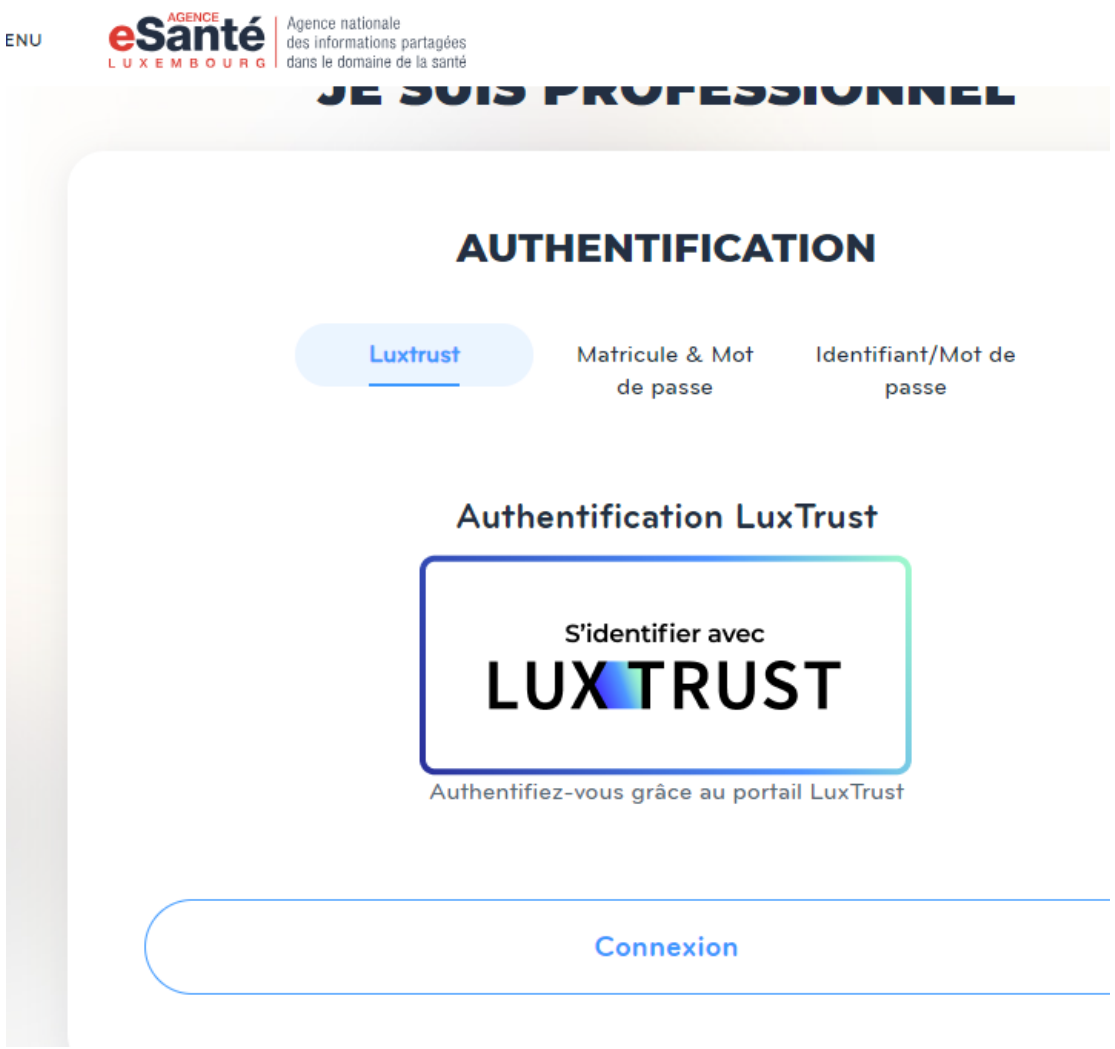
GET https://xxx.esante.lu/auth/realms/default/protocol/openid-connect/auth?
client_id=<CLIENT_ID>
&redirect_uri=<REDIRECT_URI>
&scope=openid
&response_type=code
&code_challenge=<valeur dérivée du code verifier>
&code_challenge_method=S256

URLs:
Intégration: https://www-integration.esante.lu
Production: https://www.esante.lu

2. Authentification de l'utilisateur

L'utilisateur saisit ses identifiants ou utilise son produit Luxtrust sur le portail de l'AeS.

Si OK → le serveur de l'AeS redirige le PS vers la redirect_uri définie avec un code d'autorisation qui doit être utilisé pour récupérer le token (voir slide suivant).



3. Récupération du Token

Le logiciel de médecine de ville envoie une requête POST pour échanger le code d'autorisation contre un Token :

POST https://xxx.esante.lu/auth/realms/default/protocol/openid-connect/token
grant_type=authorization_code
code=<AUTH_CODE>
redirect_uri=<REDIRECT_URI>
client_id=<CLIENT_ID>
client_secret=<CLIENT_SECRET>
code_verifier=<ORIGINAL_CODE_VERIFIER>

Réponse :
{ "refresh_token": "ey...", "access_token": "ey...", "token_type": "Bearer", "expires_in":300}

Il faut fournir l'<access_token> à l'eConnector SaaS.

Si l'<access_token> est expiré, il est possible d'en récupérer un nouveau à l'aide du <refresh_token>.

Le cycle de vie des tokens doit être géré par le logiciel de médecine de ville.

eConnector SaaS

Prérequis

- Environnement autorisant des déploiements d'image Docker.

- Récupération de l'image depuis iHUB
  - Test url :

    https://ihub-test.cns.secu.lu/downloads/eConnector-saas/1.1.6/eConnector-saas-1.1.6-integration.tar

Prérequis

Dimensionnement des machines hôtes

Environnement	vCPU	RAM	Justification
Développement / test	1 vCPU	2 Go	Suffisant pour tester la génération PDF et signature ponctuelle.
Préproduction	2 vCPU	4 Go	Permet de simuler un usage réaliste et plusieurs requêtes simultanées.
Production (charge <50 req/min)	2 vCPU	4–6 Go	Si peu d'appels simultanés.
Production (charge > 50 req/min)	4 vCPU	8 Go	Bon équilibre pour supporter plusieurs signatures/PDF en parallèle.

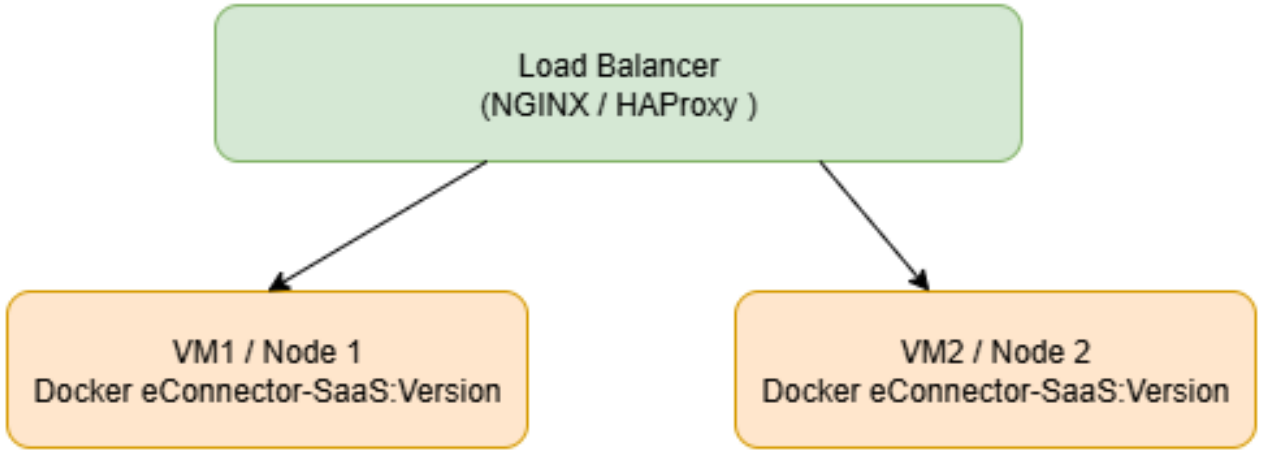
Prérequis

- Ouverture réseau sur les URL suivantes :
  - Intégration :
    - https://ihub-test.cns.secu.lu/* (service de mise à jour des templates, optionnel)
    - https://hoki-uat-tsu.incert.lu/* (service de signature)
    - https://ws.mysecu.lu:7443/* (dépôt sur mySecu)
  - Production :
    - https://ihub.cns.secu.lu/* (service de mise à jour des templates, optionnel)
    - https://hoki-tsu-kms.incert.lu/* (service de signature)
    - https://ws.mysecu.lu/* (dépôt sur mySecu)

- Enrôlement des professionnels de santé sur mySecu :
  - Signature électronique au nom du PS
  - → Obtention d'un jeton pour le PS
  - → Le PS doit être connu dans mySecu
  - URL par environnement
    - Intégration : https://mysecu-test.services-publics.lu
    - Production : https://secu.services-publics.lu/

Installation

- Faisabilité de la haute disponibilité (HA)
  - L'architecture de l'eConnector SaaS présente plusieurs avantages facilitant la mise en place d'une HA horizontale :
    - Aucune base de données requise
    - Traitements *stateless* (chaque requête est indépendante)
    - Exécution dans un conteneur Docker
    - Utilisation d'un *Spring Boot Actuator* pour la supervision et le monitoring



Installation – configuration de l'image

Variable	Description	UAT Example	Production Example
DEBUG_LOG	Enables or disables debug logs (true / false).	true	false
KMS_TIMESTAMP_URL	Timestamp service URL.	https://hoki-uat-tsu.incert.lu/api/v3/sign-timestamp	https://hoki-tsu-kms.incert.lu/api/v3/sign-timestamp
KMS_SIGN_URL	Remote signature service URL.	https://hoki-uat-tsu.incert.lu/api/v3/sign-data	https://hoki-tsu-kms.incert.lu/api/v3/sign-data
KMS_GET_CERTIFICATE_URL	Certificate retrieval URL.	https://hoki-uat-tsu.incert.lu/api/v3/get-certificate	https://hoki-tsu-kms.incert.lu/api/v3/get-certificate
TRUSTSTORE_KMS_FILENAME	Truststore filename in resource	truststore-integration.p12	truststore-prod.p12
GENERIC_KEYSTORE_FILE	Path or filename of the generic keystore (generated from the CSR fo CCSS).	keystore-integration.p12	keystore-prod.p12
GENERIC_KEYSTORE_PWD	Generic keystore password	xxxxx	xxxxx
CCSS_URL	mySecu authentication service.	https://ws.mysecu.lu:7443/ws/soap/trust?wsdl	https://ws.mysecu.lu:443/ws/soap/trust?wsdl
CCSS_SYNC_URL	mySecu business service endpoint.	https://ws.mysecu.lu:7443/ws/soap/espinst/syncexchange	https://ws.mysecu.lu:443/ws/soap/espinst/syncexchange
SYNC_EXCHANGE_WSDL	mySecu business WSDL.	https://ws.mysecu.lu:7443/ws/soap/espinst/syncexchange?wsdl	https://ws.mysecu.lu:443/ws/soap/espinst/syncexchange?wsdl
IHUB_URL	iHub update & template REST service.	https://ihub-test.cns.secu.lu/ws/rest/ihub/v1	https://ihub.cns.secu.lu/ws/rest/ihub/v1
SAAS_KEYSTORE	Keystore file that stores the server's private key and certificate, enabling secure HTTPS connections for the image.	saas-keystore-integration.p12	saas-keystore-prod.p12
SAAS_KEYSTORE_PASSWORD	Keystore password.	xxxxxx	xxxxxx
SAAS_KEYSTORE_KEY_PASSWORD	Private key password.	xxxxxx	xxxxxx
SIGNING_CA_ALIAS	Alias of the signing CA in the truststore.	test_signing_ca	signing_ca
TIMESTAMP_CA_ALIAS	Alias of the timestamp CA in the truststore.	test_timestamp_ca	timestamp_ca
ROOT_SIGNING_CA_ALIAS	Alias of the root signing CA in the truststore.	test_root_signing_ca	root_signing_ca
SERVER_PORT	Server HTTPS port (default: 8443).	8443	8443

Installation

- Lancement du conteneur et choix du profil:

  docker run -d \

  -e SPRING_PROFILES_ACTIVE=docker-integration

  -e SPRING_PROFILES_ACTIVE=docker-prod

  Le lancement de l'image nécessite l'acceptation des CGU CNS, autrement le lancement échouera.

  Vous pouvez ajouter argument suivant pour les accepter

  -e ACCEPT_LICENSE=true

  Si l'argument n'est pas présent, le lancement échoue, les CGU sont affichées dans le terminal avec le moyen de les récupérer.

NB : Ce mode de lancement est à utiliser uniquement à des fins de test.
En production, il est fortement recommandé d'utiliser un outil afin de protéger vos mots de passe et autres informations sensibles.
Pour un déploiement sécurisé et scalable, envisagez l'utilisation d'outil spécialisé pour le déploiement d'images Docker.

Utilisation de l’eConnector

- Vérification de l’état du eConnector
  - Pour s’assurer que l’image est bien déployée et que les services sont accessibles , il suffit de faire un call sur l’endpoint suivant :

    https://domain:{port}/econnector/status
  - Un retour 200 indique que l’eConnector est utilisable

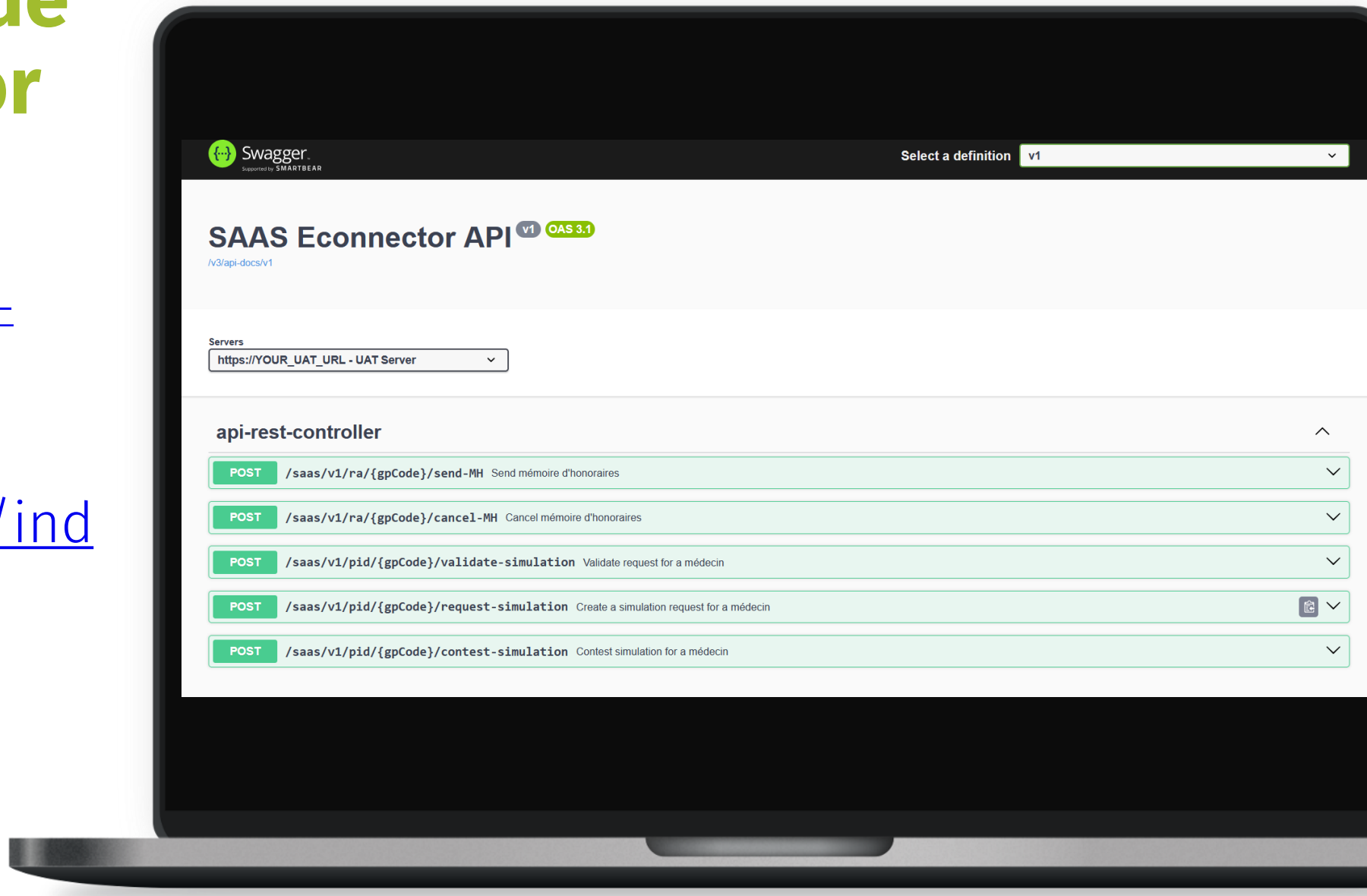
Utilisation de l’eConnector

- Api Swagger:

https://domain:{port}/swagger-ui/index.html

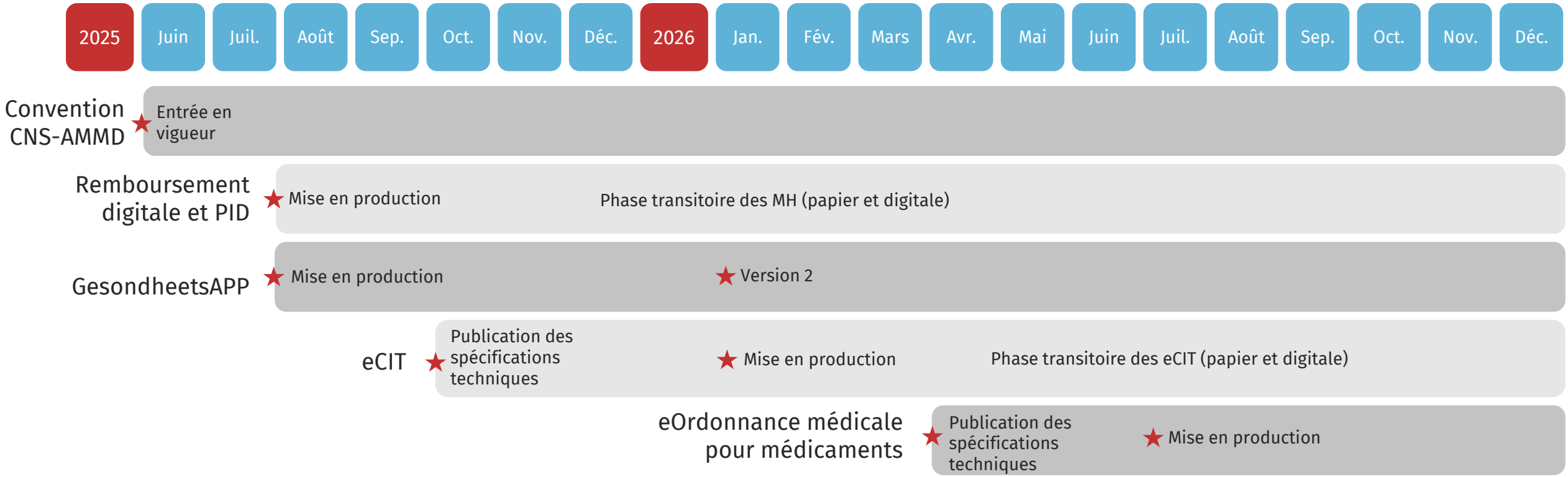
- Colletion Postaman:

https://drive.incert.lu/index.php/f/14220



Roadmap

Calendrier site web V2



PID : Paiement Immédiat Direct
MH : Mémoires d'honoraires
eCIT : Certificat d'incapacité de travail électronique

Questions/réponses

Q&A

Merci pour votre attention !

CNS
d'Gesondheetskeess