



## Documentation de l'API

Version 5.1

## Document

|                |                       |                   |            |
|----------------|-----------------------|-------------------|------------|
| Auteur         | Pierre-Emmanuel VIVER | Date de diffusion | 05/02/2025 |
| Chef de projet | Maxime REYROLLE       | N° de version     | 5.1        |

## Évolution du document

| Version | Auteur           | Nature des changements       | Date       |
|---------|------------------|------------------------------|------------|
| 1.0     | Antonio DA SILVA | Création de la documentation | 05/02/2025 |

## Licence

Ce document n'est pas libre de droits.

Ce manuel est publié sous la licence Creative Commons avec les particularités "Paternité – Partage à l'identique" (également connue sous l'acronyme CC BY-SA).

Détails de cette licence : <http://creativecommons.org/licenses/by-sa/2.0/fr/>



## Table des matières

|                                                                           |    |
|---------------------------------------------------------------------------|----|
| 1 PRÉAMBULE                                                               | 5  |
| 2 CONNEXION À LA PLATE-FORME S <sup>2</sup> LOW                           | 6  |
| 2.1 Explication                                                           | 6  |
| 2.2 Récupération de la liste des logins                                   | 6  |
| 2.3 Identification par forward de certificat                              | 6  |
| 2.4 Fonction de test de connexion                                         | 6  |
| 2.5 Fonction de récupération des informations de connexion                | 7  |
| 2.6 Fonction d'authentification par nonce                                 | 7  |
| 3 DESCRIPTION DE L'API HTTPS DE L'ADMINISTRATION DE S <sup>2</sup> LOW    | 8  |
| 3.1 Liste des groupes                                                     | 8  |
| 3.2 Liste des types de collectivités                                      | 8  |
| 3.3 Liste des collectivités                                               | 8  |
| 3.4 Liste des modules                                                     | 9  |
| 3.5 Détail d'une collectivité                                             | 9  |
| 3.6 Liste des SIREN possibles                                             | 10 |
| 3.7 Édition d'une collectivité                                            | 11 |
| 3.8 Liste des rôles des utilisateurs                                      | 12 |
| 3.9 Liste des utilisateurs                                                | 12 |
| 3.10 Détail d'un utilisateur                                              | 13 |
| 3.11 Édition d'un utilisateur                                             | 13 |
| 3.12 Ajout d'un SIREN                                                     | 14 |
| 3.13 Liste des numéros de SIRET                                           | 15 |
| 3.14 Ajout d'un numéro SIRET                                              | 15 |
| 3.15 Suppression d'un numéro SIRET                                        | 15 |
| 3.16 Ajouter un service                                                   | 16 |
| 3.17 Lister les services                                                  | 16 |
| 3.18 Ajouter un utilisateur dans un service                               | 17 |
| 4 DESCRIPTION DE L'API HTTPS POUR ACTES                                   | 18 |
| 4.1 Postage d'une transaction de transmission d'acte                      | 18 |
| 4.2 Postage d'une archive .tar.gz                                         | 19 |
| 4.3 Transaction de demande d'annulation d'acte                            | 19 |
| 4.4 Clôturer une transaction                                              | 20 |
| 4.5 Transaction de demande de classification                              | 20 |
| 4.6 Récupérer le fichier XML de la classification matières/sous-matières  | 20 |
| 4.7 Obtenir le statut d'une transaction                                   | 21 |
| 4.8 Obtenir un numéro de série pour génération enveloppe                  | 21 |
| 4.9 Définir l'URL d'archivage d'une transaction                           | 21 |
| 4.10 Récupérer la liste des documents du ministère relatifs à un ACTE     | 22 |
| 4.11 Poster une réponse à un document du ministère relatif à un ACTE      | 22 |
| 4.12 (Déprécié) Récupérer l'acte tamponné                                 | 23 |
| 4.13 Récupérer la liste des documents associés à un acte                  | 23 |
| 4.14 Récupérer l'acte (ou ses annexes), tamponné ou non                   | 24 |
| 4.15 Récupérer le bordereau d'acquittement de transaction                 | 24 |
| 4.16 Ordonner la télétransmission de l'acte Appelé par le client          | 24 |
| 4.17 Ordonner la télétransmission de plusieurs actes Appelé par le client | 25 |
| 4.18 Récupérer la liste des statuts actes possibles                       | 25 |
| 4.19 Récupérer le nombre d'actes dans un certain statut                   | 25 |
| 4.20 Récupérer les actes dans un certain statut                           | 26 |
| 4.21 Récupérer la liste de documents de la préfecture                     | 26 |
| 4.22 Marquer un document comme lu                                         | 27 |
| 4.23 Récupérer les statistiques                                           | 28 |
| 4.24 Changement du statut d'une transaction                               | 28 |
| 5 DESCRIPTION DE L'API POUR LE MODULE HÉLIOS                              | 30 |
| 5.1 Postage d'un fichier XML                                              | 30 |
| 5.2 Récupération du statut d'une transaction                              | 30 |
| 5.3 Récupération du PES ACQUIT                                            | 31 |
| 5.4 Récupération de la liste des PES_RETOUT                               | 31 |
| 5.5 Changement d'état d'un PES_RETOUT                                     | 31 |
| 5.6 Récupération du contenu d'un PES_RETOUT                               | 32 |
| 5.7 Récupérer les statistiques                                            | 32 |
| 5.8 Changement du statut d'une transaction                                | 32 |
| 6 DESCRIPTION DE L'API POUR LE MAIL SÉCURISÉ                              | 34 |
| 6.1 Version de l'API (mail sécurisé)                                      | 34 |
| 6.2 Liste des utilisateurs de la collectivité                             | 34 |
| 6.3 Nombre de mails sur le système                                        | 34 |
| 6.4 Liste de mails                                                        | 34 |
| 6.5 Détail d'un email                                                     | 35 |

|                            |    |
|----------------------------|----|
| 6.6 Envoi d'un mail        | 35 |
| 6.7 Suppression d'un email | 36 |

## 1. PRÉAMBULE

S<sup>2</sup>LOW (Service Sécurisé Libre inter-Opérable pour la Vérification et la Validation) est un logiciel de tiers de télétransmission multi-protocoles gérant les flux administratifs entre les collectivités territoriales et les administrations centrales. S<sup>2</sup>LOW implémente en premier lieu le protocole ACTES (flux de transmission aux préfectures pour le contrôle de légalité), d'autres protocoles devant venir s'ajouter progressivement (ex : HELIOS pour les transmissions au Trésor Public).

L'usage de S<sup>2</sup>LOW se fait soit via une interface Web interactive soit à travers une API programme depuis un logiciel métier.

Ce document décrit l'API HTTPS qui est utilisée par l'interface Web pour le module ACTES. Cependant, les applicatifs métiers peuvent utiliser eux-mêmes ces appels par le biais de bibliothèques génériques implémentant le protocole HTTP.

S<sup>2</sup>LOW implémente le cahier des charges ACTES sans sur-couche de transport propriétaire.

## 2. CONNEXION À LA PLATE-FORME S<sup>2</sup>LOW

### 2.1. Explication

L'application désirant utiliser cette interface devra être capable d'établir une connexion TCP vers le port 443 du serveur Web exécutant l'application S<sup>2</sup>LOW. Elle devra ensuite effectuer une négociation SSL avec authentification mutuelle.

Pour cela, elle devra présenter un certificat numérique reconnu par la plate-forme pour s'authentifier en tant qu'utilisateur attaché à une collectivité. Un compte utilisateur devra donc avoir été créé préalablement sur la plate-forme pour la détermination des droits d'accès et de l'appartenance à une collectivité. Le certificat doit avoir été importé sur le serveur S<sup>2</sup>LOW et doit ainsi être compatible avec la liste des certificats acceptés.

À partir de ce point, le dialogue HTTP peut avoir lieu au-dessus de la connexion SSL.

Des bibliothèques de programmation permettent d'effectuer ce travail fastidieux de connexion, par exemple Curl en C ou Jakarta HTTPClient en Java.

Si le certificat numérique correspond à plusieurs comptes, alors, ce certificat ne donne le droit d'utiliser qu'une seule fonction : la récupération de la liste des logins associés à ce certificat, sauf si les comptes associés utilisent un "certificat forward".

Afin de pouvoir se connecter à l'API, il convient alors d'envoyer le login et le mot de passe de l'utilisateur sous la forme d'une authentification HTTP en mode BASIC.

### 2.2. Récupération de la liste des logins

Nom du script : `/admin/users/api-list-login.php`

Méthode HTTP d'appel : GET ou POST

Description : Récupère la liste des logins associés au certificat présenté dans la requête.

Paramètres à fournir : aucun

Retour de l'appel :

- Rien s'il n'y a pas de login associé ou bien si le certificat ne correspond pas ;
- La liste des logins avec un login par ligne.

### 2.3. Identification par forward de certificat

La version 2.2 de S<sup>2</sup>LOW introduit la notion de forward de certificat. Le principe est le suivant : un certificat d'authentification sert à la connexion HTTPS. Ce certificat correspond à plusieurs comptes utilisateurs (comme avec les comptes avec des logins/mots de passe). Mais chaque compte peut être identifié via un autre certificat numérique d'identification.

Les comptes configurés avec des certificats d'identification ne sont accessibles via l'API (il n'est pas possible de se connecter sur la console). Si un compte possède à la fois un certificat d'identification et un login/mot de passe, c'est le certificat d'identification qui prime sur le login : sa présentation est obligatoire et les logins/mots de passe ne sont pas vérifiés.

Afin de présenter un certificat d'identification, il faut disposer de sa partie publique dans le format DER, l'encoder en base 64 et l'envoyer avec la requête HTTP dans un champ d'en-tête personnalisé :

`org.s2low.forward-x509-identification`

Un exemple de connexion avec PHP et la bibliothèque Curl est disponible dans le code source de S2low `test/api/connect-forward-x509.php`

### 2.4. Fonction de test de connexion

Une fonction de l'API permet de tester la connexion sans faire d'action sur S2low :

Nom du script : `/api/test-connexion.php`

Méthode HTTP d'appel : GET

Droits nécessaires : utilisateur

Description : permet de valider la connexion à S2low

Paramètres à fournir : aucun

Retour de l'appel :

- en cas de succès de l'appel : OK sur la première ligne terminée par \n
- en cas d'échec : KO sur la première ligne terminée par \n, message d'erreur sur les lignes suivantes.

## 2.5. Fonction de récupération des informations de connexion

Cette fonction est disponible à partir de la version 3.0.10 de s2Low

Cette fonction de l'API permet de récupérer les informations sur l'utilisateur qui se connecte ainsi que sur sa collectivité de rattachement.

Nom du script : `/api/info-connexion.php`

Méthode HTTP d'appel : GET

Droits nécessaires : utilisateur

Description : permet de récupérer les informations de connexion

Paramètres à fournir : aucun

Retour de l'appel :

- en cas de succès de l'appel : un tableau json contenant les informations de connexion
- en cas d'échec : KO sur la première ligne terminée par \n, message d'erreur sur les lignes suivantes.

## 2.6. Fonction d'authentification par nonce

La version 2.6.0 de S2low lève une limitation sur les fonctions appelées directement par le client en redirection d'un portail tiers : l'impossibilité d'utiliser un certificat et un login/mot de passe.

Nom du script : `/api/get-nonce.php`

Méthode HTTP d'appel : GET

Droits nécessaires : utilisateur

Description : L'appel à cette fonction est fait avec un compte associé à une collectivité. Si l'authentification réussit un nonce est envoyé (valable 5 minutes). Ce nonce permettra de raccorder un autre utilisateur à la collectivité en passant le nonce en paramètre (et donc, sans passer par la mire de login/mot de passe). Ce nonce est principalement utilisé pour les api suivantes (appelé par le client) :

- Ordonner la télétransmission de l'acte (`/actes_transac_post_confirm_api.php`)
- Ordonner la télétransmission de plusieurs actes (`/actes_transac_post_confirm_api.php`)

Paramètres à fournir : aucun

Retour : un fichier json contenant le nonce sous la forme :

```
{"nonce": "contenu du nonce"}
```

Afin de s'authentifier par la suite avec le nonce, il convient de spécifier dans les paramètres GET les informations suivantes :

| Nom paramètre | Description                                                                                                                                     |
|---------------|-------------------------------------------------------------------------------------------------------------------------------------------------|
| nonce         | le contenu du nonce retourné par get-nonce.php                                                                                                  |
| login         | le login ayant servi à la création du nonce                                                                                                     |
| hash          | le sha256 de la chaîne "password:nonce" avec password, le mot de passe associé au login précédent et nonce, le nonce retourné par get-nonce.php |

## 3. DESCRIPTION DE L'API HTTPS DE L'ADMINISTRATION DE S<sup>2</sup>LOW

### 3.1. Liste des groupes

Nom du script : `/admin/groups/admin_groups.php`

Méthode HTTP d'appel : **GET**

Droits nécessaires : Super Admin

Description : permet de lister les groupes de la plateforme.

Paramètres à fournir :

| Nom paramètre     | Présence    | Format             | Description                                                                                           |
|-------------------|-------------|--------------------|-------------------------------------------------------------------------------------------------------|
| <code>api</code>  | obligatoire | entier de valeur 1 | Indique à la plate-forme que l'appel se fait par une application métier.                              |
| <code>name</code> | facultatif  | chaîne             | Permet de filtrer la liste des groupes. Les noms des groupes récupérés contiennent la chaîne « name » |

Retour :

Une chaîne JSON contenant un tableau dont les lignes regroupent les informations sur les groupes.

| Nom paramètre       | Description                  |
|---------------------|------------------------------|
| <code>id</code>     | Identifiant unique du groupe |
| <code>name</code>   | Nom du groupe                |
| <code>status</code> | 0 : Désactivé , 1 : Activé   |

### 3.2. Liste des types de collectivités

Nom du script : `/admin/authorities/admin_authority_types.php`

Méthode HTTP d'appel : **GET**

Droits nécessaires : Super admin ou administrateur de groupe

Description : permet de lister les types de collectivités

Paramètres à fournir : Aucun

Retour :

Une chaîne JSON contenant un tableau contenant des lignes de type id => description

| Nom paramètre            | Description                                 |
|--------------------------|---------------------------------------------|
| <code>id</code>          | Identifiant interne du type de collectivité |
| <code>description</code> | Description du type de la collectivité      |

### 3.3. Liste des collectivités

Nom du script : `/admin/authorities/admin_authorities.php`

Méthode HTTP d'appel : **GET**

Droits nécessaires : Super Admin ou administrateur de groupe

Description : permet de lister les collectivités de la plateforme.

Paramètres à fournir :

| Nom paramètre       | Présence    | Format             | Description                                                                                                                                                                                                                                                |
|---------------------|-------------|--------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>api</code>    | obligatoire | entier de valeur 1 | Indique à la plate-forme que l'appel se fait par une application métier.                                                                                                                                                                                   |
| <code>name</code>   | facultatif  | chaîne             | Permet de filtrer la liste des collectivités. Les noms des groupes récupérés contiennent la chaîne « name »                                                                                                                                                |
| <code>type</code>   | facultatif  | chaîne             | Filtre suivant un des identifiants de type de collectivité retourné par la fonction <code>admin_authority_types.php</code>                                                                                                                                 |
| <code>groupe</code> | facultatif  | chaîne             | Filtre les collectivités qui appartiennent au groupe dont l'identifiant est donné par ce paramètre. Ce paramètre n'est utilisable que pour le super admin. Les identifiants de groupe peuvent être récupérés via la fonction <code>admin_groups.php</code> |
| <code>siren</code>  | facultatif  | Entier             | Filtre sur tout ou partie d'un SIREN.                                                                                                                                                                                                                      |

Retour :

Une chaîne JSON contenant un tableau dont les lignes regroupent les informations sur les collectivités.

| Nom paramètre                   | Description                                                                             |
|---------------------------------|-----------------------------------------------------------------------------------------|
| <code>id</code>                 | Identifiant unique de la collectivité                                                   |
| <code>name</code>               | Nom de la collectivité                                                                  |
| <code>authority_group_id</code> | Identifiant du groupe de collectivité (voir la fonction <code>admin_groups.php</code> ) |

Autres paramètres de retour : `siren` , `address` , `postal_code` , `city` , `telephone` , `fax`

### 3.4. Liste des modules

Nom du script : `/admin/modules/admin_modules.php`

Méthode HTTP d'appel : GET

Droits nécessaires : Super admin ou administrateur de groupe

Description : permet de lister les modules actifs

Paramètres à fournir :

| Nom paramètre    | Présence    | Format             | Description                                                              |
|------------------|-------------|--------------------|--------------------------------------------------------------------------|
| <code>api</code> | obligatoire | entier de valeur 1 | Indique à la plate-forme que l'appel se fait par une application métier. |

Retour :

Une chaîne JSON contenant un tableau contenant des lignes décrivant un module

| Nom paramètre            | Description                   |
|--------------------------|-------------------------------|
| <code>id</code>          | Identifiant interne du module |
| <code>description</code> | Description du module         |
| <code>name</code>        | Nom du module                 |

### 3.5. Détail d'une collectivité

Nom du script : `/admin/authorities/admin_authority_detail.php`

Méthode HTTP d'appel : GET

Droits nécessaires : Super Admin, Admin de groupe, ou Admin de collectivité

Description : permet d'obtenir l'ensemble des informations sur une collectivité

Paramètres à fournir :

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

| Nom paramètre | Présence    | Format | Description                    |
|---------------|-------------|--------|--------------------------------|
| id            | obligatoire | entier | Identifiant de la collectivité |

Retour :

Une chaîne JSON contenant un tableau associatif contenant les informations sur la collectivité.

| Nom paramètre           | Description                                                                            |
|-------------------------|----------------------------------------------------------------------------------------|
| id                      | Identifiant unique de la collectivité                                                  |
| name                    | Nom de la collectivité                                                                 |
| status                  | 0 : Désactivé , 1 : Activé                                                             |
| authority_type_id       | Identifiant de type de collectivité retourné par la fonction admin_authority_types.php |
| siren                   | SIREN de la collectivité                                                               |
| email                   | Email de contact de la collectivité                                                    |
| postal_code             | Code Postal                                                                            |
| city                    | Ville                                                                                  |
| department              | Département (sur trois chiffres)                                                       |
| district                | Arrondissement (1 chiffre)                                                             |
| telephone               | Téléphone                                                                              |
| fax                     | Fax                                                                                    |
| default_broadcast_email | Email de notification sélectionné par défaut lors de la création d'un ACTE             |
| broadcast_email_address | Email de notification non-sélectionné par défaut lors de la création d'un ACTE         |
| email_mail_securise     | Email de provenance des mails sécurisés (champ « from »)                               |
| sae_wsdl                | URL du WSDL du SAE (Asal@e)                                                            |
| sae_login               | Login du SAE                                                                           |
| sae_password            | Mot de passe du SAE                                                                    |
| sae_id_versant          | SEA Identifiant du service versant                                                     |
| sae_id_archive          | SAE Identifiant service d'archive                                                      |
| sae_originating_agency  | SAE Identifiant service producteur                                                     |
| sae_numero_aggrement    | SAE Accord de versement                                                                |

Les paramètres suivants ne sont disponibles que pour le super Admin ou un administrateur de groupe.

| Nom paramètre       | Description                                                                                                                                                                                    |
|---------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| authority_group_id  | Identifiant du groupe dont fait partie la collectivité                                                                                                                                         |
| helios_ftp_login    | HELIOS ftp login                                                                                                                                                                               |
| helios_ftp_password | HELIOS ftp mot de passe                                                                                                                                                                        |
| helios_ftp_dest     | HELIOS ftp Dest                                                                                                                                                                                |
| ext_siret           | Numéro de EXT SIRET                                                                                                                                                                            |
| module              | Tableau associatif de type clé => Valeur avec les clés, les identifiants de module retournés par la fonction admin_module.php et comme valeur :<br>0 : module désactivé ,<br>1 : module activé |

### 3.6. Liste des SIREN possibles

Nom du script : `/admin/authorities/admin_authorities_siren.php`

Méthode HTTP d'appel : GET

Droits nécessaires : Super admin ou administrateur de groupe

Description : permet de lister les SIREN autorisés pour créer une collectivité

Paramètres à fournir :

| Nom paramètre                   | Présence                                                         | Format | Description           |
|---------------------------------|------------------------------------------------------------------|--------|-----------------------|
| <code>authority_group_id</code> | Obligatoire pour le super admin. Interdit pour l'admin de groupe | entier | Identifiant du groupe |

Retour : Une chaîne JSON contenant un tableau contenant la liste des SIREN autorisés.

### 3.7. Édition d'une collectivité

Nom du script : `/admin/authorities/admin_authority_edit_handler.php`

Méthode HTTP d'appel : POST

Droits nécessaires : Super admin, administrateur de groupe ou administrateur de collectivité

Description : permet de créer ou de modifier une collectivité

Paramètres à fournir :

Les lignes grisées indiquent les lignes qui ne peuvent être modifiées ou créées que par le super admin ou l'administrateur de groupe.

| Nom paramètre                        | Présence    | Format             | Description                                                                                        |
|--------------------------------------|-------------|--------------------|----------------------------------------------------------------------------------------------------|
| <code>api</code>                     | obligatoire | entier de valeur 1 | Indique à la plate-forme que l'appel se fait par une application métier.                           |
| <code>name</code>                    | obligatoire | chaîne             | Nom de la collectivité                                                                             |
| <code>siren</code>                   | obligatoire | numéro SIREN       | numéro SIREN autorisé pour ce groupe (voir fonction <code>admin_authorities_siren.php</code> )     |
| <code>authority_groupe_id</code>     | obligatoire | entier             | Identifiant du groupe auquel est rattachée cette collectivité. Facultatif pour un admin de groupe. |
| <code>email</code>                   | facultatif  | chaîne             | email de la collectivité                                                                           |
| <code>default_broadcast_email</code> | facultatif  | chaîne             | email de notification coché par défaut pour ACTES (séparé par des virgules)                        |
| <code>broadcast_email</code>         | facultatif  | chaîne             | Autres emails de notification (séparés par des virgules)                                           |
| <code>status</code>                  | obligatoire | 0 ou 1             | 0 : désactivé , 1 : activé                                                                         |
| <code>authority_type_id</code>       | obligatoire | entier             | Identifiant du type de collectivité (voir la fonction <code>admin_authority_types.php</code> )     |
| <code>address</code>                 | facultatif  | chaîne             | Adresse de la collectivité                                                                         |
| <code>postal_code</code>             | facultatif  | chaîne             | Code Postal                                                                                        |
| <code>city</code>                    | facultatif  | chaîne             | Ville                                                                                              |
| <code>department</code>              | obligatoire | chaîne             | Département sur trois chiffres (ex : 069 ; 972; 02B)                                               |
| <code>district</code>                | obligatoire | entier             | numéro de l'arrondissement (sur un chiffre)                                                        |
| <code>telephone</code>               | facultatif  | chaîne             | Numéro de téléphone                                                                                |
| <code>fax</code>                     | facultatif  | chaîne             | Numéro de fax                                                                                      |
| <code>ext_siret</code>               | facultatif  | chaîne             | Numéro SIRET                                                                                       |
| <code>helios_ftp_login</code>        | facultatif  | chaîne             | Login FTP pour HELIOS                                                                              |
| <code>helios_ftp_password</code>     | facultatif  | chaîne             | Mot de passe FTP pour HELIOS                                                                       |
| <code>helios_ftp_dest</code>         | facultatif  | chaîne             | HELIOS ftp Dest                                                                                    |
| <code>email_mail_securise</code>     | facultatif  | chaîne             | Email source des mails sécurisés émis par la collectivité.                                         |
| <code>perm_XXX</code>                | obligatoire | 0 ou 1             | Indique si le module XXX est activé. Voir la fonction <code>admin_modules.php</code> .             |

Remarque : Il est nécessaire de fournir à chaque fois l'ensemble des paramètres pour une modification, ceux-ci peuvent être obtenus grâce à la fonction `admin_authority_detail.php`

Retour :

Une chaîne JSON contenant un status :

- **error** : dans ce cas, msg-erreur contient le message de l'erreur.
- **ok** : dans ce cas, id contient l'identifiant de la collectivité créée ou modifiée.

### 3.8. Liste des rôles des utilisateurs

Nom du script : `/admin/users/admin_users_role.php`

Méthode HTTP d'appel : GET

Droits nécessaires : Super admin ou administrateur de groupe ou administrateur de collectivité

Description : permet de lister les rôles des utilisateurs

Paramètres à fournir : Aucun

Retour :

Une chaîne JSON contenant un tableau contenant des lignes de type id => description

| Nom paramètre      | Description                 |
|--------------------|-----------------------------|
| <b>id</b>          | Identifiant interne du rôle |
| <b>description</b> | Description du rôle         |

### 3.9. Liste des utilisateurs

Nom du script : `/admin/users/admin_users.php`

Méthode HTTP d'appel : GET

Droits nécessaires : Super Admin ou administrateur de groupe ou administrateur de collectivité

Description : permet de lister les collectivités de la plateforme.

Paramètres à fournir :

| Nom paramètre    | Présence    | Format             | Description                                                                                                                                                                                                                                                                 |
|------------------|-------------|--------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>api</b>       | obligatoire | entier de valeur 1 | Indique à la plate-forme que l'appel se fait par une application métier.                                                                                                                                                                                                    |
| <b>name</b>      | facultatif  | chaîne             | Permet de filtrer la liste des utilisateurs. Les noms des utilisateurs récupérés contiennent la chaîne « name »                                                                                                                                                             |
| <b>role</b>      | facultatif  | chaîne             | Filtre suivant un des identifiants de rôle retourné par la fonction <code>admin_users_role.php</code>                                                                                                                                                                       |
| <b>group</b>     | facultatif  | entier             | Filtre les utilisateurs dont la collectivité appartient au groupe dont l'identifiant est donné par ce paramètre. Ce paramètre n'est utilisable que pour le super admin. Les identifiants de groupe peuvent être récupérés via la fonction « <code>admin_groups.php</code> » |
| <b>authority</b> | facultatif  | entier             | Filtre les utilisateurs appartenant à la collectivité dont l'identifiant est donné par ce paramètre. Ne fonctionne pas pour un admin de collectivité.                                                                                                                       |

Retour :

Une chaîne JSON contenant un tableau dont les lignes regroupent les informations sur les utilisateurs.

| Nom paramètre | Description                         |
|---------------|-------------------------------------|
| <b>id</b>     | Identifiant unique de l'utilisateur |
| <b>name</b>   | Nom de l'utilisateur                |

|                             |                                                                       |
|-----------------------------|-----------------------------------------------------------------------|
| <code>givenname</code>      | Prénom                                                                |
| <code>email</code>          | email de l'utilisateur                                                |
| <code>role</code>           | Rôle de l'utilisateur, voir la fonction <code>admin_users_role</code> |
| <code>status</code>         | 0 : désactivé , 1 : activé                                            |
| <code>authority_id</code>   | Identifiant de la collectivité à laquelle appartient l'utilisateur    |
| <code>authority_name</code> | Nom de la collectivité                                                |

### 3.10. Détail d'un utilisateur

Nom du script : `/admin/users/admin_user_detail.php`

Méthode HTTP d'appel : GET

Droits nécessaires : Super Admin, Admin de groupe, ou Admin de collectivité

Description : permet d'obtenir l'ensemble des informations sur un utilisateur

Paramètres à fournir :

| Nom paramètre   | Présence    | Format | Description                  |
|-----------------|-------------|--------|------------------------------|
| <code>id</code> | obligatoire | entier | Identifiant de l'utilisateur |

Retour :

Une chaîne JSON contenant un tableau associatif contenant les informations sur l'utilisateur.

| Nom paramètre                   | Description                                                                                                                                                                                                                                          |
|---------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>id</code>                 | Identifiant unique de l'utilisateur                                                                                                                                                                                                                  |
| <code>name</code>               | Nom                                                                                                                                                                                                                                                  |
| <code>givenname</code>          | Prénom                                                                                                                                                                                                                                               |
| <code>login</code>              | Login de l'utilisateur (si il y a plusieurs utilisateurs pour le même certificat)                                                                                                                                                                    |
| <code>email</code>              | email de l'utilisateur                                                                                                                                                                                                                               |
| <code>telephone</code>          | Téléphone de l'utilisateur                                                                                                                                                                                                                           |
| <code>status</code>             | 0 : Désactivé , 1 : Activé                                                                                                                                                                                                                           |
| <code>authority_id</code>       | Identifiant de la collectivité de l'utilisateur                                                                                                                                                                                                      |
| <code>authority_group_id</code> | Identifiant du groupe dont fait partie la collectivité de l'utilisateur                                                                                                                                                                              |
| <code>role</code>               | Rôle de l'utilisateur (voir <code>admin_users_role.php</code> )                                                                                                                                                                                      |
| <code>certificate</code>        | Le contenu du certificat de l'utilisateur au format PEM                                                                                                                                                                                              |
| <code>module</code>             | Tableau associatif avec comme clé l'identifiant des modules (voir <code>admin_modules.php</code> ) et comme valeur une des valeurs suivantes :<br>- NONE : pas de droit ,<br>- RO : lecture seule ,<br>- RW : Modification ,<br>- GRANT : concession |
| <code>other_id</code>           | Tableau contenant les autres identifiants d'utilisateur qui partagent ce certificat                                                                                                                                                                  |

### 3.11. Édition d'un utilisateur

Nom du script : `/admin/users/admin_user_edit_handler.php`

Méthode HTTP d'appel : POST

Droits nécessaires : Super admin ou administrateur de groupe ou administrateur de collectivité

Description : permet de créer ou de modifier les propriétés d'un utilisateur.

Paramètres à fournir :

| Nom paramètre                   | Présence    | Format             | Description                                                                                                                                                                                                                                                                                                                                              |
|---------------------------------|-------------|--------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>api</code>                | obligatoire | entier de valeur 1 | Indique à la plate-forme que l'appel se fait par une application métier.                                                                                                                                                                                                                                                                                 |
| <code>id</code>                 |             | entier             | Identifiant de l'utilisateur à modifier. Il doit être renseigné dans l'URL. Si aucun, il s'agit de la création d'un utilisateur                                                                                                                                                                                                                          |
| <code>name</code>               | obligatoire | chaîne             | Nom de l'utilisateur                                                                                                                                                                                                                                                                                                                                     |
| <code>givenname</code>          | obligatoire | chaîne             | Prénom                                                                                                                                                                                                                                                                                                                                                   |
| <code>email</code>              | facultatif  | chaîne             | Email                                                                                                                                                                                                                                                                                                                                                    |
| <code>telephone</code>          | facultatif  | chaîne             | Téléphone                                                                                                                                                                                                                                                                                                                                                |
| <code>status</code>             | facultatif  | 0 ou 1             | 0 : désactivé , 1 : activé                                                                                                                                                                                                                                                                                                                               |
| <code>authority_id</code>       | obligatoire | entier             | Identifiant de la collectivité. Ce paramètre n'est pas modifiable une fois l'utilisateur créé. Il dépend également des droits de l'utilisateur qui utilise l'API (il ne sert à rien pour un admin de collectivité)                                                                                                                                       |
| <code>role</code>               | obligatoire | chaîne             | Un des rôles suivants : SADM (super admin), GADM (admin de groupe), ADM (admin de collectivité), USER (utilisateur). Il n'est bien sûr pas possible de créer un utilisateur avec un rôle dont on ne dispose pas soi-même.                                                                                                                                |
| <code>authority_group_id</code> | obligatoire | entier             | Identifiant du groupe. Ne sert que pour la création d'un utilisateur par le super admin. Ne peut pas être modifié.                                                                                                                                                                                                                                       |
| <code>certificate</code>        | obligatoire | fichier PEM        | Fichier contenant le certificat de l'utilisateur. Obligatoire lors d'une création (sauf à utiliser <code>new_id</code> , voir plus loin). Facultatif lors d'une modification.                                                                                                                                                                            |
| <code>login</code>              | facultatif  | chaîne             | Login de l'utilisateur pour un partage de certificat. Obligatoire dans ce cas.                                                                                                                                                                                                                                                                           |
| <code>password</code>           | facultatif  | chaîne             | Mot de passe correspondant.                                                                                                                                                                                                                                                                                                                              |
| <code>new_id</code>             | facultatif  | entier             | Permet de copier un certificat pour lui attribuer un nouvel utilisateur. <code>new_id</code> représente l'identifiant de l'utilisateur dont on veut copier le certificat. Cet utilisateur, comme celui qu'on va créer doivent impérativement disposer chacun d'un login/mot de passe                                                                     |
| <code>perm_XXX</code>           | facultatif  | chaîne             | XXX représente un numéro de module (voir <code>admin_modules.php</code> ) et comme valeur une des valeurs suivantes : NONE (pas de droit), RO (lecture seule), RW (Modification), GRANT (concession). La possibilité de modifier les paramètres dépend de nos propres droits et des droits de la collectivité ou du groupe à utiliser tel ou tel module. |

Remarque : Il est nécessaire de fournir à chaque fois l'ensemble des paramètres pour une modification, ceux-ci peuvent être obtenus grâce à la fonction `admin_users_detail.php`

Retour :

Une chaîne JSON contenant un status :

- `error` : dans ce cas, error-message contient le message de l'erreur.
- `ok` : dans ce cas, id contient l'identifiant de l'utilisateur créé ou modifié.

## 3.12. Ajout d'un SIREN

Nom du script : `/admin/authorities/admin_authorities_add_siren.php`

Méthode HTTP d'appel : GET

Droits nécessaires : Super admin ou administrateur de groupe

Description : permet d'ajouter un siren dans un groupe

Paramètres à fournir :

| Nom paramètre | Présence | Format | Description |
|---------------|----------|--------|-------------|
|---------------|----------|--------|-------------|

|                                 |             |                      |                                                                      |
|---------------------------------|-------------|----------------------|----------------------------------------------------------------------|
| <code>authority_group_id</code> | obligatoire | entier               | Correspond à l'id du groupe dans lequel on souhaite ajouter le siren |
| <code>siren</code>              | obligatoire | entier de 9 chiffres | SIREN à ajouter dans le groupe                                       |

Retour :

Une chaîne JSON contenant un status :

- `error` : dans ce cas, error-message contient le message de l'erreur.(Echec de l'authentification, Accès refusé, identifiant groupe invalide, siren non valide, siren déjà présent)
- `ok` : dans ce cas, message contient « ajout réussi »

### 3.13. Liste des numéros de SIRET

Nom du script : `/admin/authorities/admin_authority_siret.php`

Méthode HTTP d'appel : GET

Droits nécessaires : Super admin ou administrateur de groupe

Description : permet de lister les numéros de SIRET à une collectivité.

Paramètres à fournir :

| Nom paramètre      | Présence    | Format                | Description                                                              |
|--------------------|-------------|-----------------------|--------------------------------------------------------------------------|
| <code>api</code>   | obligatoire | entier de valeur 1    | Indique à la plate-forme que l'appel se fait par une application métier. |
| <code>siret</code> | obligatoire | entier de 14 chiffres | Numéro SIRET                                                             |

Retour :

Une chaîne JSON contenant :

- `error` : dans ce cas, error-message contient le message de l'erreur.(Echec de l'authentification, Accès refusé, identifiant groupe invalide, siren non valide, siren déjà présent), ou
- la liste des SIRET de la collectivité

### 3.14. Ajout d'un numéro SIRET

Nom du script : `/admin/authorities/admin_authority_siret_add.php`

Méthode HTTP d'appel : POST

Droits nécessaires : Super admin ou administrateur de groupe

Description : permet d'ajouter un numéro de SIRET à une collectivité.

Paramètres à fournir :

| Nom paramètre             | Présence    | Format                | Description                                                              |
|---------------------------|-------------|-----------------------|--------------------------------------------------------------------------|
| <code>api</code>          | obligatoire | entier de valeur 1    | Indique à la plate-forme que l'appel se fait par une application métier. |
| <code>authority_id</code> | obligatoire | entier                | Identifiant de la collectivité                                           |
| <code>siret</code>        | obligatoire | entier de 14 chiffres | Numéro SIRET                                                             |

Retour :

Une chaîne JSON contenant un status :

- `error` : dans ce cas, error-message contient le message de l'erreur.(Echec de l'authentification, Accès refusé, identifiant groupe invalide, siren non valide, siren déjà présent)
- `ok` : dans ce cas, message contient « Numéro de SIRET ajouté »

### 3.15. Suppression d'un numéro SIRET

Nom du script : `/admin/authorities/admin_authority_siret_del.php`

Méthode HTTP d'appel : POST

Droits nécessaires : Super admin ou administrateur de groupe

Description : permet de supprimer un numéro de SIRET à une collectivité.

Paramètres à fournir :

| Nom paramètre | Présence    | Format                | Description                                                              |
|---------------|-------------|-----------------------|--------------------------------------------------------------------------|
| api           | obligatoire | entier de valeur 1    | Indique à la plate-forme que l'appel se fait par une application métier. |
| id            | obligatoire | entier                | Identifiant de la collectivité                                           |
| siret         | obligatoire | entier de 14 chiffres | Numéro SIRET                                                             |

Retour :

Une chaîne JSON contenant un status :

- **error** : dans ce cas, error-message contient le message de l'erreur.(Echec de l'authentification, Accès refusé, identifiant groupe invalide, siren non valide, siren déjà présent)
- **ok** : dans ce cas, message contient « Numéro de SIRET retiré »

### 3.16. Ajouter un service

Cette fonction est disponible à partir de la version 3.0.13 de S<sup>2</sup>LOW

Nom du script : `admin/services/add-service-user.php`

Méthode HTTP d'appel : POST

Droits nécessaires : administrateur (super, groupe ou collectivité)

Description : permet d'ajouter un service à une collectivité

Paramètres à fournir :

| Nom paramètre | Présence    | Format                           | Description                                                              |
|---------------|-------------|----------------------------------|--------------------------------------------------------------------------|
| api           | obligatoire | entier de valeur 1               | Indique à la plate-forme que l'appel se fait par une application métier. |
| authority_id  | obligatoire | entier                           | Identifiant de la collectivité                                           |
| name          | obligatoire | chaîne de 128 caractères maximum | Nom du service à ajouter                                                 |

Retour :

Une chaîne JSON contenant un status :

- **error** : dans ce cas, error-message contient le message de l'erreur.(Echec de l'authentification, Accès refusé, service déjà existant)
- **ok**

### 3.17. Lister les services

Cette fonction est disponible à partir de la version 3.0.13 de S<sup>2</sup>LOW

Nom du script : `admin/services/list-service.php`

Méthode HTTP d'appel : GET

Droits nécessaires : administrateur (super, groupe ou collectivité)

Description : Liste les services associés à une collectivité

Paramètres à fournir :

| Nom paramètre | Présence | Format | Description |
|---------------|----------|--------|-------------|
|               |          |        |             |

|                           |             |        |                                |
|---------------------------|-------------|--------|--------------------------------|
| <code>authority_id</code> | obligatoire | entier | Identifiant de la collectivité |
|---------------------------|-------------|--------|--------------------------------|

Retour :

Une chaîne JSON contenant un status :

- `error` : dans ce cas, error-message contient le message de l'erreur.(Echec de l'authentification, Accès refusé)
- ou bien un fichier json contenant les informations sur les services (id = l'identifiant du service)

### 3.18. Ajouter un utilisateur dans un service

Cette fonction est disponible à partir de la version 3.0.13 de S²LOW

Nom du script : `admin/users/add-user-to-service.php`

Méthode HTTP d'appel : POST

Droits nécessaires : administrateur (super, groupe ou collectivité)

Description : Ajoute un utilisateur dans un service

Paramètres à fournir :

| Nom paramètre           | Présence    | Format             | Description                                                              |
|-------------------------|-------------|--------------------|--------------------------------------------------------------------------|
| <code>api</code>        | obligatoire | entier de valeur 1 | Indique à la plate-forme que l'appel se fait par une application métier. |
| <code>id_service</code> | obligatoire | entier             | Identifiant du service retourné par la fonction list-service.php         |
| <code>id_user</code>    | obligatoire | entier             | Identifiant de l'utilisateur                                             |

Retour :

Une chaîne JSON contenant un status :

- `error` : dans ce cas, error-message contient le message de l'erreur.(Echec de l'authentification, Accès refusé, service déjà existant)
- `ok`

## 4. DESCRIPTION DE L'API HTTPS POUR ACTES

L'API HTTPS consiste en des scripts PHP qui sont appelés par des méthodes HTTP GET ou POST.

Tous les chemins de scripts indiqués par la suite sont relatifs à l'adresse de base du serveur (qui est de la forme `https://nom.du.serveur`).

Certains scripts ont besoin de savoir que l'appel est effectué par une application métier afin d'adapter la réponse retournée dans un format plus facilement exploitable par une application (par défaut les scripts retournent des redirections HTTP vers une page qui affiche le résultat du traitement). C'est à cette fin que ces scripts acceptent dans les requêtes un paramètre nommé « `api` » qui prend pour valeur « 1 ». Si ce paramètre n'est pas présent ou initialisé à une autre valeur, le résultat retourné est formaté pour un client navigateur et non une application métier.

Dans la suite, nous décrivons chaque service avec son script API, la méthode et les paramètres, ainsi que le retour du serveur.

### 4.1. Postage d'une transaction de transmission d'acte

Nom du script : `/modules/actes/actes_transac_create.php`

Méthode HTTP d'appel : `POST` avec `enctype="multipart/form-data"`

Description : Soumission d'un acte à transmettre au ministère. Nécessite la fourniture de toutes les informations concernant l'acte à transmettre. La plate-forme se charge de générer l'archive .tar.gz et de la poster au ministère.

Paramètres à fournir :

| Nom paramètre                        | Présence       | Format                                                  | Description                                                                                                                                                                         |
|--------------------------------------|----------------|---------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>api</code>                     | obligatoire    | entier de valeur 1                                      | Indique à la plate-forme que l'appel se fait par une application métier. Change le type de retour fourni par TédéTIS.                                                               |
| <code>nature_code</code>             | obligatoire    | entier                                                  | Code nature de l'acte issu de la classification matière.                                                                                                                            |
| <code>classif1</code>                | obligatoire    | entier                                                  | Code classification niveau 1                                                                                                                                                        |
| <code>classif2</code>                | obligatoire    | entier                                                  | Code classification niveau 2                                                                                                                                                        |
| <code>classif3</code>                | facultative    | entier                                                  | Code classification niveau 3                                                                                                                                                        |
| <code>classif4</code>                | facultative    | entier                                                  | Code classification niveau 4                                                                                                                                                        |
| <code>classif5</code>                | facultative    | entier                                                  | Code classification niveau 5                                                                                                                                                        |
| <code>number</code>                  | obligatoire    | 15 caractères maxi, chiffres, lettres en majuscule ou _ | Numéro de l'acte interne à la collectivité                                                                                                                                          |
| <code>decision_date</code>           | obligatoire    | date ANSI : YYYY-MM-DD                                  | Date de décision de l'acte                                                                                                                                                          |
| <code>subject</code>                 | obligatoire    | texte longueur maxi 500 caractères                      | Objet de l'acte                                                                                                                                                                     |
| <code>acte_pdf_file</code>           | obligatoire    | fichier attaché                                         | Fichier PDF ou XML de l'acte                                                                                                                                                        |
| <code>acte_pdf_file_sign</code>      | facultative    | fichier attaché                                         | Fichier de signature de l'acte au format PKCS7                                                                                                                                      |
| <code>type_acte</code>               | obligatoire(1) | 5 caractères                                            | Type de pièce jointe, un des codes qui apparaît dans les attributs CodeTypePJ de la classification                                                                                  |
| <code>acte_attachments[]</code>      | facultative    | fichiers attachés                                       | Fichiers de pièces jointes (PDF, JPG ou PNG). Ce paramètre peut apparaître plusieurs fois.                                                                                          |
| <code>acte_attachments_sign[]</code> | facultative    | fichiers attachés                                       | Fichiers de signature des pièces jointes. Ce paramètre peut apparaître plusieurs fois.                                                                                              |
| <code>type_pj[]</code>               | obligatoire(1) | 5 caractères                                            | Type de pièce jointe, un des codes qui apparaît dans les attributs CodeTypePJ de la classification                                                                                  |
| <code>en_attente</code>              | facultative    | entier de valeur 0 ou 1                                 | 0 (valeur par défaut) : la transaction sera envoyée en préfecture , 1 : la transaction est placée dans un état d'attente. Un opérateur devra la valider pour envoi à la préfecture. |
|                                      |                |                                                         | Permet à la collectivité d'indiquer si la télétransmission est multi-canal,                                                                                                         |

|                 |             |                                  |                                                                                          |
|-----------------|-------------|----------------------------------|------------------------------------------------------------------------------------------|
| document_papier | facultative | entier de valeur 0 (défaut) ou 1 | c'est-à-dire avec des documents complémentaires envoyés au format papier (1 dans ce cas) |
|-----------------|-------------|----------------------------------|------------------------------------------------------------------------------------------|

**Remarque :** Les deux variables `acte_attachments[]` et `acte_attachments_sign[]` sont vues comme des tableaux par la plate-forme et doivent être synchronisées. Le premier élément du tableau `acte_attachments_sign[]` doit être le fichier contenant la signature du premier élément du tableau `acte_attachments[]` et ainsi de suite. (1) **Remarque :** À compter du 08/06/2019, la typologie de la pièce principale et des annexes est obligatoire

Retour de l'appel :

- en cas de succès de l'appel : OK sur la première ligne terminée par `\n`, numéro de la transaction créée sur la deuxième ligne
- en cas d'échec : KO sur la première ligne terminée par `\n`, message d'erreur sur les lignes suivantes.

**Déprécié :** l'appel avec la méthode `GET` ne sera plus possible dans une prochaine version

## 4.2. Postage d'une archive .tar.gz

Nom du script : `/modules/actes/actes_transac_submit.php`

Méthode HTTP d'appel : `POST` avec `enctype="multipart/form-data"`

Description : Soumission d'une archive .tar.gz contenant une ou plusieurs transactions à transmettre au ministère. Nécessite la fourniture d'un fichier .tar.gz. La plate-forme se charge de valider l'archive et de la poster au ministère.

Paramètres à fournir :

| Nom paramètre | Présence    | Format             | Description                                                                                                           |
|---------------|-------------|--------------------|-----------------------------------------------------------------------------------------------------------------------|
| api           | obligatoire | entier de valeur 1 | Indique à la plate-forme que l'appel se fait par une application métier. Change le type de retour fourni par TédÉTIS. |
| enveloppe     | obligatoire | fichier attaché    | Fichier archive .tar.gz contenant les transactions                                                                    |

Retour de l'appel :

- en cas de succès de l'appel : OK sur la première ligne terminée par `\n`, nombre de transactions créées sur la deuxième ligne terminée par `\n`, numéros des transactions créées sur les lignes suivantes (dans le même ordre que dans l'enveloppe XML de l'archive)
- en cas d'échec : KO sur la première ligne terminée par `\n`, message d'erreur sur les lignes suivantes.

## 4.3. Transaction de demande d'annulation d'acte

Nom du script : `/modules/actes/actes_transac_cancel.php`

Méthode HTTP d'appel : `POST`

Description : Génération d'une transaction de demande d'annulation d'un acte.

Paramètres à fournir :

| Nom paramètre | Présence    | Format             | Description                                                                                                           |
|---------------|-------------|--------------------|-----------------------------------------------------------------------------------------------------------------------|
| api           | obligatoire | entier de valeur 1 | Indique à la plate-forme que l'appel se fait par une application métier. Change le type de retour fourni par TédÉTIS. |
| id            | obligatoire | entier             | Numéro d'identifiant de la transaction à annuler (numéro retourné lors de la création d'un acte)                      |

Retour de l'appel :

- en cas de succès de l'appel : OK sur la première ligne terminée par `\n`, numéro de la transaction d'annulation créée sur la deuxième ligne
- en cas d'échec : KO sur la première ligne terminée par `\n` message d'erreur sur les lignes suivantes.

**Déprécié :** l'appel avec la méthode `GET` ne sera plus possible dans une prochaine version. L'usage de cette API reste possible en version 5.0.

## 4.4. Clôturer une transaction

Nom du script : `/modules/actes/actes_transac_close.php`

Méthode HTTP d'appel : `POST`

Description : Passer une transaction de transmission d'acte dans l'état « Validé » ou « Annulé ».

Paramètres à fournir :

| Nom paramètre       | Présence    | Format             | Description                                                                                                                                   |
|---------------------|-------------|--------------------|-----------------------------------------------------------------------------------------------------------------------------------------------|
| <code>api</code>    | obligatoire | entier de valeur 1 | Indique à la plate-forme que l'appel se fait par une application métier. Change le type de retour fourni par TédÉTIS.                         |
| <code>id</code>     | obligatoire | entier             | Numéro d'identifiant de la transaction à clôturer (numéro retourné lors de la création d'un acte)                                             |
| <code>status</code> | obligatoire | chaîne             | Statut dans lequel passer la transaction : <code>valid</code> : passer en état « Validé » ; <code>invalid</code> : passer en état « Refusé ». |

Retour de l'appel :

- en cas de succès de l'appel : OK sur la première ligne terminée par `\n`.
- en cas d'échec : KO sur la première ligne terminée par `\n`, message d'erreur sur les lignes suivantes.

Déprécié : l'appel avec la méthode `GET` ne sera plus possible dans une prochaine version

## 4.5. Transaction de demande de classification

Nom du script : `/modules/actes/actes_classification_request.php`

Méthode HTTP d'appel : `POST`

Description : Génération d'une transaction de demande de classification matières/sous-matières.

Paramètres à fournir :

| Nom paramètre    | Présence    | Format             | Description                                                                                                           |
|------------------|-------------|--------------------|-----------------------------------------------------------------------------------------------------------------------|
| <code>api</code> | obligatoire | entier de valeur 1 | Indique à la plate-forme que l'appel se fait par une application métier. Change le type de retour fourni par TédÉTIS. |

Retour de l'appel :

- en cas de succès de l'appel : OK sur la première ligne terminée par `\n`, numéro de la transaction de demande de classification créée sur la deuxième ligne.
- en cas d'échec : KO sur la première ligne terminée par `\n`, message d'erreur sur les lignes suivantes.

Déprécié : l'appel avec la méthode `GET` ne sera plus possible dans une prochaine version

## 4.6. Récupérer le fichier XML de la classification matières/sous-matières

Nom du script : `/modules/actes/actes_classification_fetch.php`

Méthode HTTP d'appel : `GET`

Description : Récupérer le fichier XML de la dernière classification matières/sous-matières pour la collectivité.

Paramètres à fournir :

| Nom paramètre    | Présence    | Format             | Description                                                                                                           |
|------------------|-------------|--------------------|-----------------------------------------------------------------------------------------------------------------------|
| <code>api</code> | obligatoire | entier de valeur 1 | Indique à la plate-forme que l'appel se fait par une application métier. Change le type de retour fourni par TédÉTIS. |

Retour de l'appel :

- en cas de succès de l'appel : retour du fichier XML en attachement (nom de fichier : classification.xml)
- en cas d'échec : KO sur la première ligne terminée par `\n`, message d'erreur sur les lignes suivantes.

## 4.7. Obtenir le statut d'une transaction

Nom du script : `/modules/actes/actes_transac_get_status.php`

Méthode HTTP d'appel : `GET`

Description : Obtention du code de statut d'une transaction.

Paramètres à fournir :

| Nom paramètre            | Présence    | Format | Description                              |
|--------------------------|-------------|--------|------------------------------------------|
| <code>transaction</code> | obligatoire | entier | Identifiant de la transaction concernée. |

Retour de l'appel :

- en cas de succès de l'appel : OK sur la première ligne terminée par `\n`, statut de la transaction sur la deuxième ligne. Liste des statuts :
- -1 : Erreur ;
- 0 : Annulé ;
- 1 : Posté ;
- 2 : En attente de transmission ;
- 3 : Transmis ;
- 4 : Acquittance reçue ;
- 5 : Validé ;
- 6 : Refusé.

Pour le statut 4, les lignes suivantes contiennent le contenu du fichier d'acquittance (ARActe). Pour le statut -1, la troisième ligne contient le message d'erreur.

- en cas d'échec : KO sur la première ligne terminée par `\n`, message d'erreur sur les lignes suivantes.

## 4.8. Obtenir un numéro de série pour génération enveloppe

Nom du script : `/modules/actes/actes_transac_get_env_serial.php`

Méthode HTTP d'appel : `GET`

Description : Obtention d'un numéro de série unique pour génération d'une enveloppe.

Paramètres à fournir :

- Aucun

Retour de l'appel :

- en cas de succès de l'appel : OK sur la première ligne terminée par `\n`, numéro de série unique sur la deuxième ligne
- en cas d'échec : KO sur la première ligne terminée par `\n`, message d'erreur sur les lignes suivantes.

## 4.9. Définir l'URL d'archivage d'une transaction

Nom du script : `/modules/actes/actes_transac_set_archive_url.php`

Méthode HTTP d'appel : `POST`

Description : Définition de l'URL d'archivage d'une transaction.

Paramètres à fournir :

| Nom paramètre   | Présence    | Format | Description                              |
|-----------------|-------------|--------|------------------------------------------|
| <code>id</code> | obligatoire | entier | Identifiant de la transaction concernée. |

|     |             |                    |                                                                                                                       |
|-----|-------------|--------------------|-----------------------------------------------------------------------------------------------------------------------|
| url | obligatoire | chaîne             | URL d'archivage de la transaction                                                                                     |
| api | obligatoire | entier de valeur 1 | Indique à la plate-forme que l'appel se fait par une application métier. Change le type de retour fourni par TédÉTIS. |

Retour de l'appel :

- en cas de succès de l'appel : OK sur la première ligne terminée par `\n` ;
- en cas d'échec : KO sur la première ligne terminée par `\n` , message d'erreur sur les lignes suivantes.

## 4.10. Récupérer la liste des documents du ministère relatifs à un ACTE

Nom du script : `/modules/actes/actes_transac_get_document.php`

Méthode HTTP d'appel : `GET`

Description : Définition de l'URL d'archivage d'une transaction.

Paramètres à fournir :

| Nom paramètre | Présence    | Format | Description                              |
|---------------|-------------|--------|------------------------------------------|
| id            | obligatoire | entier | Identifiant de la transaction concernée. |

Retour de l'appel :

- en cas de succès de l'appel : Liste de l'ensemble des documents relatifs à l'acte OU BIEN le document lui-même si la transaction est une transaction de document relatif.

La liste contient une ligne par document avec trois numéros représentant :

- le type de document (2, 3, 4, 5, 6 )
  - 2 : Courrier simple ;
  - 3 : Demande de pièces complémentaires ;
  - 4 : Lettre d'observation ;
  - 5 : Délégué au tribunal administratif ;
  - 6 : Annulation (il ne s'agit pas d'un message provenant de la préfecture)/
- le statut du document (7, 8 ou 21 pour les documents reçus, 1 à 5 pour les documents envoyés) :
  - 7 : document reçu ;
  - 8 : document reçu et acquittement envoyé ;
  - 21 : document reçu ne nécessitant pas d'acquittement ;
  - Les numéros 1 à 5 ont la même signification que pour la transmission d'un acte original.
- l'identifiant interne de la transaction (numérique) ;
- en cas d'échec : KO sur la première ligne terminée par `\n` , message d'erreur sur les lignes suivantes.

## 4.11. Poster une réponse à un document du ministère relatif à un ACTE

Nom du script : `/modules/actes/actes_transac_reponse_create.php`

Méthode HTTP d'appel : `POST`

Description : Définition de l'URL d'archivage d'une transaction.

Paramètres à fournir :

| Nom paramètre | Présence                                                                                   | Format             | Description                                                                                                           |
|---------------|--------------------------------------------------------------------------------------------|--------------------|-----------------------------------------------------------------------------------------------------------------------|
| id            | obligatoire                                                                                | entier             | Identifiant de la transaction concernée.                                                                              |
| api           | obligatoire                                                                                | entier de valeur 1 | Indique à la plate-forme que l'appel se fait par une application métier. Change le type de retour fourni par TédÉTIS. |
| type_envoie   | Obligatoire pour les messages de type « Pièce complémentaire » ou « Lettre d'observation » | Entier ( 3 ou 4)   | 3 : Refus d'envoi de pièce , 4 : envoi de pièce                                                                       |

|                                 |                                                            |                   |                                                                                                                                                    |
|---------------------------------|------------------------------------------------------------|-------------------|----------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>acte_pdf_file</code>      | obligatoire                                                | fichiers attachés | Fichier PDF de la réponse                                                                                                                          |
| <code>type_acte</code>          | obligatoire(1)                                             | 5 caractères      | Type de pièce jointe de <code>acte_pdf_file</code> , un des codes qui apparaît dans les attributs <code>CodeTypePJ</code> de la classification     |
| <code>acte_attachments[]</code> | Facultative uniquement pour les « Pièces complémentaires » | Fichiers attachés | Fichiers de pièces jointes (PDF, JPG ou PNG). Ce paramètre peut apparaître plusieurs fois.                                                         |
| <code>type_pj</code>            | obligatoire(1)                                             | 5 caractères      | Type de pièce jointe des <code>acte_attachments</code> , un des codes qui apparaît dans les attributs <code>CodeTypePJ</code> de la classification |

Retour de l'appel :

- en cas de succès de l'appel : OK sur la première ligne terminée par `\n` les documents relatifs à l'acte sur les lignes suivantes.
- en cas d'échec : KO sur la première ligne terminée par `\n`, message d'erreur sur les lignes suivantes.

## 4.12. (Déprécié) Récupérer l'acte tamponné

Cette fonction est dépréciée dans la version 2.0 de S<sup>2</sup>LOW. Il n'est plus certain qu'elle soit disponible dans les versions ultérieures de S<sup>2</sup>LOW. Il convient d'utiliser :

- `actes_transac_get_files_list.php` pour récupérer la liste des fichiers attachés à un acte ;
- `actes_download_file.php?file=id&tampon=true` pour récupérer le fichier tamponné.

Nom du script : `/modules/actes/actes_transac_get_tampon.php`

Méthode HTTP d'appel : `GET`

Description : Récupération du fichier acte, tamponné par S<sup>2</sup>low

Paramètres à fournir :

| Nom paramètre            | Présence    | Format | Description                              |
|--------------------------|-------------|--------|------------------------------------------|
| <code>transaction</code> | obligatoire | entier | Identifiant de la transaction concernée. |

Retour de l'appel :

- en cas de succès de l'appel : l'acte tamponné ;
- en cas d'échec : KO sur la première ligne terminée par `\n`, message d'erreur sur les lignes suivantes.

## 4.13. Récupérer la liste des documents associés à un acte

Nom du script : `/modules/actes/actes_transac_get_files_list.php`

Méthode HTTP d'appel : `GET`

Description : Récupération de la liste des fichiers associés à un acte (fichier métier, l'acte lui-même et les annexes)

Paramètres à fournir :

| Nom paramètre            | Présence    | Format | Description                              |
|--------------------------|-------------|--------|------------------------------------------|
| <code>transaction</code> | obligatoire | entier | Identifiant de la transaction concernée. |

Retour de l'appel :

- en cas de succès de l'appel : un fichier JSON contenant un tableau des documents disponibles.  
Chaque document contient :
  - un id (permettant de récupérer le document, cf, fonction suivante) ;
  - name : le nom original du fichier ;
  - posted\_filename : le nom du fichier, tel que posté par l'utilisateur (si posté par l'utilisateur) ;
  - mimetype : le type de contenu ;
  - size : la taille en octet ;
  - signature : la signature détachée au format PKCS#7 si disponible.

On notera que le premier fichier est toujours le fichier métier XML, le second est l'acte lui-même et les suivants, les annexes attachées à l'acte.

- en cas d'échec : KO sur la première ligne terminée par `\n`, message d'erreur sur les lignes suivantes.

## 4.14. Récupérer l'acte (ou ses annexes), tamponné ou non

Nom du script : `/modules/actes/actes_download_file.php`

Méthode HTTP d'appel : `GET`

Description : Récupération un fichier attaché à un acte.

Paramètres à fournir :

| Nom paramètre               | Présence               | Format               | Description                                                                                           |
|-----------------------------|------------------------|----------------------|-------------------------------------------------------------------------------------------------------|
| <code>file</code>           | obligatoire            | entier               | Identifiant du fichier tel que retourné par la fonction <code>actes_transac_get_files_list.php</code> |
| <code>tampon</code>         | non (false par défaut) | Boolean              | Indique si le fichier doit être tamponné (ne fonctionne que pour les fichiers au format PDF)          |
| <code>date_affichage</code> | non                    | Date au format Y-m-d | Ajoute la date de publication (anciennement date d'affichage) de l'acte dans le tampon                |

Retour de l'appel :

- le contenu du fichier lui-même.
- une erreur 404 peut-être retournée en cas d'échec.

## 4.15. Récupérer le bordereau d'acquittement de transaction

Nom du script : `/modules/actes/actes_create_pdf.php`

Méthode HTTP d'appel : `GET`

Description : Récupération du document (format PDF) bordereau d'acquittement de transaction.

Paramètres à fournir :

| Nom paramètre         | Présence    | Format | Description                              |
|-----------------------|-------------|--------|------------------------------------------|
| <code>trans_id</code> | obligatoire | entier | Identifiant de la transaction concernée. |

Retour de l'appel :

- le contenu du fichier lui-même.

## 4.16. Ordonner la télétransmission de l'acte Appelé par le client

Nom du script : `/modules/actes/actes_transac_post_confirm_api.php`

Méthode HTTP d'appel : `GET`

Description : Ordonne la télétransmission pour un acte qui est dans l'état « en attente d'être posté. » Cette fonction permet à un site tiers de préparer la télétransmission, puis de la faire télétransmettre en direct par un utilisateur sur S2low. On peut par exemple utiliser cela pour permettre à l'utilisateur de présenter un certificat RGS\*\* pour assurer la télétransmission.

Paramètres à fournir :

| Nom paramètre           | Présence    | Format | Description                                                                                                                                                                                                                                                     |
|-------------------------|-------------|--------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>id</code>         | obligatoire | entier | Identifiant de la transaction.                                                                                                                                                                                                                                  |
| <code>url_return</code> | obligatoire | URL    | Lors de la terminaison de cette fonction, on redirige l'utilisateur vers cette url. L'url peut contenir : <code>%%ERROR %%</code> qui est remplacé par 0 en cas de réussite et 1 en cas d'erreur. <code>%%MESSAGE %%</code> qui contiendra le message d'erreur. |

Retour de l'appel : N/A

## 4.17. Ordonner la télétransmission de plusieurs actes Appelé par le client

Nom du script : `/modules/actes/actes_transac_post_confirm_api_multi.php`

Méthode HTTP d'appel : `GET`

Description : Ordonne la télétransmission pour un acte qui est dans l'état « en attente d'être posté. » Cette fonction permet à un site tiers de préparer la télétransmission, puis de la faire télétransmettre en direct par un utilisateur sur S2low. On peut par exemple utiliser cela pour permettre à l'utilisateur de présenter un certificat RGS\*\* pour assurer la télétransmission.

Paramètres à fournir :

| Nom paramètre           | Présence    | Format | Description                                                                                                                                                                                                                    |
|-------------------------|-------------|--------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>id[]</code>       | obligatoire | entier | Tableau d'identifiant des transactions.                                                                                                                                                                                        |
| <code>url_return</code> | obligatoire | URL    | Lors de la terminaison de cette fonction, on redirige l'utilisateur vers cette url. Aucune erreur n'est générée ni retournée par la fonction. C'est au système appelant de vérifier que ces actes ont été télétransmis ou non. |

Retour de l'appel : N/A

## 4.18. Récupérer la liste des statuts actes possibles

Nom du script : `/modules/actes/api/actes_status.php`

Méthode HTTP d'appel : `GET`

Description : Récupère les statuts des actes disponibles sur la plateforme

Paramètres à fournir : aucun

Retour de l'appel : un fichier JSON contenant la liste des status (identifiant du statut et libellé)

Exemple de retour :

```
{
  "-1": "Erreur",
  "0": "Annul\u00e9",
  "1": "Post\u00e9",
  "2": "En attente de transmission",
  "3": "Transmis",
  "4": "Acquittement re\u00e7u",
  "5": "Valid\u00e9",
  "6": "Refus\u00e9",
  "7": "Document re\u00e7u",
  "8": "Acquittement envoy\u00e9",
  "9": "Document envoy\u00e9",
  "10": "Refus d'envoi",
  "11": "Acquittement de document re\u00e7u",
  "12": "Envoy\u00e9 au SAE",
  "13": "Archiv\u00e9 par le SAE",
  "14": "Erreur lors de l'archivage",
  "15": "Re\u00e7u par le SAE",
  "16": "D\u00e9truite",
  "17": "En attente d'\u00eatre post\u00e9",
  "18": "En attente d'\u00eatre sign\u00e9",
  "19": "En attente de transmission au SAE",
  "20": "Erreur lors de l'envoi au SAE",
  "21": "Document re\u00e7u (pas d'AR)"
}
```

## 4.19. Récupérer le nombre d'actes dans un certain statut

Nom du script : `/modules/actes/api/number_actes.php`

Méthode HTTP d'appel : `GET`

Description : Récupère le nombre d'actes dans un certain status en fonction de sa collectivité de rattachement

Paramètres à fournir :

| Nom paramètre          | Présence                  | Format | Description                                  |
|------------------------|---------------------------|--------|----------------------------------------------|
| <code>status_id</code> | facultatif (0 par défaut) | entier | Identifiant du statut (cf actes_status.php). |

Retour de l'appel : un fichier JSON contenant les informations de la requête et le nombre de transactions dans le statut sélectionné.

Exemple de retour :

```
{
  "status_id": 4,
  "authority_id": 2,
  "nb_transactions": 80
}
```

## 4.20. Récupérer les actes dans un certain statut

Nom du script : `/modules/actes/api/list_actes.php`

Méthode HTTP d'appel : `GET`

Description : Récupère les actes dans un certain statut en fonction de sa collectivité de rattachement

Paramètres à fournir :

| Nom paramètre          | Présence                       | Format            | Description                                  |
|------------------------|--------------------------------|-------------------|----------------------------------------------|
| <code>status_id</code> | facultatif (0 par défaut)      | entier            | Identifiant du statut (cf actes_status.php). |
| <code>offset</code>    | facultatif (0 par défaut)      | entier            | Identifiant du premier élément à récupérer   |
| <code>limit</code>     | facultatif (100 par défaut)    | entier            | Nombre de transaction à récupérer            |
| <code>min_date</code>  | facultatif (aucune par défaut) | date (YYYY-mm-dd) | Date du passage en statut le plus ancien     |
| <code>max_date</code>  | facultatif (aucune par défaut) | date (YYYY-mm-dd) | Date du passage en statut le plus récent     |

Retour de l'appel : un fichier JSON contenant les informations sur les transactions actes dans le statut sélectionné.

Exemple de retour :

```
{
  "status_id": "4",
  "authority_id": "2",
  "offset": "0",
  "limit": "2",
  "transactions": [
    {
      "id": "158",
      "subject": "45456",
      "number": "201709111149",
      "date": "2017-09-01",
      "nature_descr": "Actes individuels",
      "classification": "3.5",
      "type": "1"
    },
    {
      "id": "157",
      "subject": "5465",
      "number": "201709111135",
      "date": "2017-09-01",
      "nature_descr": "Actes réglementaires",
      "classification": "2.3",
      "type": "1"
    }
  ]
}
```

## 4.21. Récupérer la liste de documents de la préfecture

Cette fonction est disponible à partir de la version 3.0.15

Nom du script : `/modules/actes/api/list_document_prefecture.php`

Méthode HTTP d'appel : `GET`

Description : Permet de lister l'ensemble des documents non-lu envoyés par la préfecture (message 2-1, 3-1, 4-1 et 5-1). Il faut marquer les transactions récupérer comme lu afin de ne pas les récupérer à nouveau avec la fonction **marquer un document comme lu**

Paramètres à fournir : aucun

Retour de l'appel : un fichier JSON contenant la liste des transactions

| Nom de la clé                       | Description                                                                            |
|-------------------------------------|----------------------------------------------------------------------------------------|
| <code>id</code>                     | Identifiant de la transaction.                                                         |
| <code>type</code>                   | type du message selon la norme actes (exemple : 3 = demande de pièces complémentaires) |
| <code>related_transaction_id</code> | Identifiant de la transaction original (message 1-1) de s2low concerné par le message  |
| <code>number</code>                 | Numéro interne de l'acte (message 1-1)                                                 |
| <code>unique_id</code>              | Numéro unique de l'acte (message 1-1)                                                  |
| <code>last_status_id</code>         | Status de la transaction envoyé par la préfecture                                      |

Exemple de retour :

```
[
  {
    "id": "56",
    "type": "3",
    "related_transaction_id": "55",
    "number": "201805281136",
    "unique_id": "002-000000000-20180501-201805281136-AR",
    "last_status_id": "8"
  },
  {
    "id": "103",
    "type": "3",
    "related_transaction_id": "102",
    "number": "201808201712",
    "unique_id": "02A-000000000-20180801-201808201712-AI",
    "last_status_id": "8"
  },
  {
    "id": "106",
    "type": "2",
    "related_transaction_id": "105",
    "number": "201808211106",
    "unique_id": "02A-000000000-20180801-201808211106-AI",
    "last_status_id": "21"
  },
  {
    "id": "107",
    "type": "3",
    "related_transaction_id": "105",
    "number": "201808211106",
    "unique_id": "02A-000000000-20180801-201808211106-AI",
    "last_status_id": "8"
  },
  {
    "id": "108",
    "type": "5",
    "related_transaction_id": "105",
    "number": "201808211106",
    "unique_id": "02A-000000000-20180801-201808211106-AI",
    "last_status_id": "21"
  },
  {
    "id": "109",
    "type": "4",
    "related_transaction_id": "105",
    "number": "201808211106",
    "unique_id": "02A-000000000-20180801-201808211106-AI",
    "last_status_id": "8"
  }
]
```

## 4.22. Marquer un document comme lu

Cette fonction est disponible à partir de la version 3.0.15

Nom du script : `/modules/actes/api/document_prefecture_mark_as_read.php`

Méthode HTTP d'appel : `GET`

Description : Permet de marquer un document de la préfecture comme lu pour ne pas le récupérer à nouveau via la fonction **récupérer la liste de documents de la préfecture**

Paramètres à fournir :

| Nom paramètre               | Présence    | Format | Description                    |
|-----------------------------|-------------|--------|--------------------------------|
| <code>transaction_id</code> | obligatoire | entier | Identifiant de la transaction. |

Retour de l'appel : un fichier JSON contenant le résultat (OK)

Exemple de retour :

```
{"result": "ok"}
```

## 4.23. Récupérer les statistiques

Cette fonction est disponible à partir de la version 4.3.9

Nom du script : `/modules/actes/api/nb_actes_by_authorities_and_date.php`

Méthode HTTP d'appel : `GET`

Description : Permet à un profil admin de récupérer le nombre d'actes par autorité pour un groupe et un mois donné. Le mois est considéré à partir du 01-month-year.

Paramètres à fournir :

| Nom paramètre                   | Présence   | Format | Description                                                              |
|---------------------------------|------------|--------|--------------------------------------------------------------------------|
| <code>authority_group_id</code> | facultatif | entier | Identifiant du groupe d'autorité. (par défaut : groupe de l'utilisateur) |
| <code>month</code>              | facultatif | entier | Mois. (par défaut : mois précédent)                                      |
| <code>year</code>               | facultatif | entier | Année. (par défaut : année du mois précédent)                            |

Retour de l'appel : un fichier JSON contenant le résultat (OK)

Exemple de retour :

```
{
  "result": "ok",
  "message": "",
  "authority_group_id": 1,
  "min_date": "2021-6-01",
  "max_date": "2021-06-30",
  "nbTransactionPerAuthorities": [
    {
      "authority_id": 2,
      "name": "Test",
      "nb_transactions": 10
    }
  ]
}
```

## 4.24. Changement du statut d'une transaction

Nom du script : `/modules/actes/api/actes_sae_status.php`

Méthode HTTP d'appel : `POST`

Droits nécessaires : archiviste

Description : Permet à un utilisateur archiviste de changer le statut d'une transaction pour qu'elle termine en état archivé sur s2low.

Paramètres à fournir dans le corps de la requête :

| Nom paramètre               | Présence    | Format | Description                    |
|-----------------------------|-------------|--------|--------------------------------|
| <code>transaction_id</code> | obligatoire | entier | Identifiant de la transaction  |
| <code>status_id</code>      | obligatoire | entier | Identifiant du statut souhaité |

Seuls certains états sources et cibles permettent le changement de statut.

La transaction doit se trouver dans un de ces états :

- `4` : Acquittement reçu
- `5` : Validé

- 12 : Envoyé au SAE
- 14 : Erreur lors de l'archivage

La liste des états cible est la suivante :

- 12 : Envoyé au SAE
- 13 : Archivé par le SAE
- 14 : Erreur lors de l'archivage

Retour de l'appel :

- un fichier JSON contenant le résultat
- un code HTTP 200 en succès et 400 en erreur

Exemples de retour :

- En cas de succès :

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

- En cas d'erreur :

```
{  
  "error": "Transition vers le statut 4 impossible"  
}
```

## 5. DESCRIPTION DE L'API POUR LE MODULE HÉLIOS

### 5.1. Postage d'un fichier XML

Nom du script : `/modules/helios/api/helios_importer_fichier.php`

Méthode HTTP d'appel : **POST** avec enctype="multipart/formdata"

Description : Soumission d'un fichier XML représentant le document à transmettre à un comptable. Nécessite la fourniture d'un fichier .xml. La plateforme se charge de valider le fichier et de le poster au destinataire.

Paramètres à fournir :

| Nom paramètre          | Présence    | Format          | Description                                      |
|------------------------|-------------|-----------------|--------------------------------------------------|
| <code>enveloppe</code> | obligatoire | fichier attaché | Fichier XML qui représente le document financier |

Retour de l'appel (fichier XML) :

```
<import>
<id> numéro de la transaction créée </id>
<resultat> OK ou KO </resultat>
<message> message complémentaire </message>
</import>
```

**Attention** : une signature invalide donnera lieu à un OK pour notifier la création de la transaction, mais cette dernière apparaîtra immédiatement en statut erreur (1).

### 5.2. Récupération du statut d'une transaction

Nom du script : `/modules/helios/api/helios_transac_get_status.php`

Méthode HTTP d'appel : **GET**

Description : Obtention du code de statut d'une transaction.

Paramètres à fournir :

| Nom paramètre            | Présence    | Format | Description                             |
|--------------------------|-------------|--------|-----------------------------------------|
| <code>transaction</code> | obligatoire | entier | Identifiant de la transaction concernée |

Retour de l'appel : fichier xml

```
<transaction>
<id> numéro de la transaction </id>
<resultat> OK ou KO </resultat>
<status> statut </status>
<message> message complémentaire </message>
</transaction>
```

Liste des statuts :

| Code | Description                |
|------|----------------------------|
| -1   | Erreur                     |
| 1    | Posté                      |
| 2    | En attente de transmission |
| 3    | Transmis                   |
| 4    | Ack                        |
| 6    | Nack                       |
| 7    | En traitement              |
| 8    | Information disponible     |
| 9    | Envoyé au SAE              |
| 10   | Accepter par le SAE        |

|    |                                   |
|----|-----------------------------------|
| 11 | Refuser par le SAE                |
| 13 | En attente d'être signée          |
| 14 | En attente d'être postée          |
| 19 | En attente de transmission au SAE |
| 20 | Erreur lors de l'envoi au SAE     |

### 5.3. Récupération du PES ACQUIT

Nom du script : `/modules/helios/api/helios_download_acquit.php`

Méthode HTTP d'appel : `GET`

Description : Récupération de l'acquittement PES ACQUIT correspondant à un PES ALLER

Paramètres à fournir :

| Nom paramètre   | Présence    | Format | Description                              |
|-----------------|-------------|--------|------------------------------------------|
| <code>id</code> | obligatoire | entier | Identifiant de la transaction concernée. |

Retour de l'appel : Directement le fichier PES ACQUIT

### 5.4. Récupération de la liste des PES\_RETOUT

Nom du script : `/modules/helios/api/helios_get_list.php`

Méthode HTTP d'appel : `GET`

Description : Obtention de la liste des PES\_RETOUT avec l'état « non lu »

Paramètres à fournir :

| Nom paramètre             | Présence    | Format | Description                               |
|---------------------------|-------------|--------|-------------------------------------------|
| <code>collectivite</code> | obligatoire | entier | Identifiant de la collectivité concernée. |

Retour de l'appel : fichier xml

```
<liste>
<idColl> identifiant de la collectivité </idColl>
4/5
Documentation API HTTPS - TédÉTIS Ed1.1.0
<dateDemande> date de demande de la liste </dateDemande>
<resultat> OK ou KO </resultat>
<message> message complémentaire </message>
<pes_retour>
<id> identifiant du pes_retour </id>
<nom> nom </nom>
<date> date de réception du pes_retour par le tdt </date>
</pes_retour>
<pes_retour>
<id> identifiant du pes_retour </id>
<nom> nom </nom>
<date> date de réception du pes_retour par le tdt </date>
</pes_retour>
...
</liste>
```

### 5.5. Changement d'état d'un PES\_RETOUT

Nom du script : `/modules/helios/api/helios_change_status.php`

Méthode HTTP d'appel : `GET`

Description : Changement d'état de PES\_RETOUT : passage de l'état non lu à l'état lu.

Paramètres à fournir :

| Nom paramètre   | Présence    | Format | Description                          |
|-----------------|-------------|--------|--------------------------------------|
| <code>id</code> | obligatoire | entier | Identifiant des pes_retour concernés |

Retour de l'appel : fichier xml

```
<change>
<id> numéro de la transaction </id>
<resultat> OK ou KO </resultat>
<message> message complémentaire </message>
</change>
```

## 5.6. Récupération du contenu d'un PES\_RETOUR

Nom du script : `/modules/helios/api/helios_get_retour.php`

Méthode HTTP d'appel : `GET`

Description : Récupération du fichier xml correspondant au pes\_retour demandé.

Paramètres à fournir :

| Nom paramètre   | Présence    | Format | Description                          |
|-----------------|-------------|--------|--------------------------------------|
| <code>id</code> | obligatoire | entier | Identifiant des pes_retour concernés |

Retour de l'appel : le fichier xml du pes\_retour

## 5.7. Récupérer les statistiques

Cette fonction est disponible à partir de la version 4.3.9

Nom du script : `/modules/helios/api/nb_pes_aller_by_authorities_and_date.php`

Méthode HTTP d'appel : `GET`

Description : Permet à un profil admin de récupérer le nombre de pes aller par autorité pour un groupe et un mois donné. Le mois est considéré à partir du 01-month-year.

Paramètres à fournir :

| Nom paramètre                   | Présence   | Format | Description                                                              |
|---------------------------------|------------|--------|--------------------------------------------------------------------------|
| <code>authority_group_id</code> | facultatif | entier | Identifiant du groupe d'autorité. (par défaut : groupe de l'utilisateur) |
| <code>month</code>              | facultatif | entier | Mois. (par défaut : mois précédent)                                      |
| <code>year</code>               | facultatif | entier | Année. (par défaut : année du mois précédent)                            |

Retour de l'appel : un fichier JSON contenant le résultat (OK)

Exemple de retour :

```
{
  "result": "ok",
  "message": "",
  "authority_group_id": 1,
  "min_date": "2021-6-01",
  "max_date": "2021-06-30",
  "nbTransactionPerAuthorities": [
    {
      "authority_id": 2,
      "name": "Test",
      "nb_transactions": 10
    }
  ]
}
```

## 5.8. Changement du statut d'une transaction

Nom du script : `/modules/helios/helios_transac_change_status_sae.php`

Méthode HTTP d'appel : `POST`

Droits nécessaires : archiviste

Description : Permet à un utilisateur archiviste de changer le statut d'une transaction pour qu'elle termine en état accepté par le SAE sur

s2low.

Paramètres à fournir dans le corps de la requête :

| Nom paramètre  | Présence    | Format             | Description                                                              |
|----------------|-------------|--------------------|--------------------------------------------------------------------------|
| api            | obligatoire | entier de valeur 1 | Indique à la plate-forme que l'appel se fait par une application métier. |
| transaction_id | obligatoire | entier             | Identifiant de la transaction                                            |
| status_id      | obligatoire | entier             | Identifiant du statut souhaité                                           |

Seuls certains états sources et cibles permettent le changement de statut.

- 8 : Information disponible
  - 19 : En attente de transmission SAE
- 19 : En attente de transmission SAE
  - 10 : Accepté par le SAE
  - 20 : Erreur lors de l'envoi SAE
- 20 : Erreur lors de l'envoi SAE
  - 19 : En attente de transmission SAE
  - 10 : Accepté par le SAE

Retour de l'appel :

- un fichier JSON contenant le résultat

Exemples de retour :

- En cas de succès :

```
{
  "status": "ok",
  "message": "Le status de la transaction a été modifiée"
}
```

- En cas d'erreur :

```
{
  "status": "error",
  "error-message": "Impossible de changer le status de la transaction"
}
```

## 6. DESCRIPTION DE L'API POUR LE MAIL SÉCURISÉ

Cette partie décrit les fonctions de l'API HTTP permettant d'accéder via un programme au module de mail sécurisé du logiciel S2low.

Tous les chemins de scripts indiqués dans cette partie sont relatifs à l'adresse de base du serveur (qui est de la forme `https://nom.du.serveur`).

L'ensemble des scripts est situé dans le répertoire `/modules/mail/api/`

### 6.1. Version de l'API (mail sécurisé)

Nom du script : `/modules/mail/api/version.php`

Méthode HTTP d'appel : `GET`

Description : Renvoie la version de l'API mail sécurisé

Paramètres à fournir : aucun

Retour : Un fichier texte ligne à ligne (exemple ci-dessous).

```
version_api
detail_version_api
```

### 6.2. Liste des utilisateurs de la collectivité

Nom du script : `/modules/mail/api/user-mail.php`

Méthode HTTP d'appel : `GET`

Paramètres à fournir : aucun

Retour : Un fichier texte ligne à ligne ou chaque ligne désigne une entrée de l'annuaire. Les lignes sont formatées de la manière suivante :

```
id_utilisateur:email:description[:groupe[:groupe ...]]
```

### 6.3. Nombre de mails sur le système

Nom du script : `/modules/mail/api/nb-mail.php`

Méthode HTTP d'appel : `GET`

Paramètres à fournir : aucun

Retour : Le nombre de mails (entier)

### 6.4. Liste de mails

Nom du script : `/modules/mail/api/list-mail.php`

Méthode HTTP d'appel : `GET`

Paramètres à fournir :

| Nom paramètre       | Présence    | Format | Description                          |
|---------------------|-------------|--------|--------------------------------------|
| <code>limit</code>  | facultative | entier | Nombre maximum de mails listés       |
| <code>offset</code> | facultative | entier | Sauter offset mail avant l'affichage |

Retour :

Le script affiche **limit** mails classés par date inverse de création à partir de l'email d'ordre **offset**.

Chaque ligne représente les informations sur un email de la manière suivante :

```
id_mail:date creation:état confirmation:objet
```

`état confirmation` est soit :

- aucune confirmation,
- confirmé partiellement,
- confirmé

## 6.5. Détail d'un email

Nom du script : ``/modules/mail/api/detail-mail.php`

Méthode HTTP d'appel : `GET`

Paramètre du script :

| Nom paramètre   | Présence    | Format | Description                                              |
|-----------------|-------------|--------|----------------------------------------------------------|
| <code>id</code> | obligatoire | entier | Identifiant du mail retourné par le script list-mail.php |

Retour : un fichier contenant l'ensemble des informations sur le mail

```
id : *****
date_envoi : *****
password : *****
fn_download : *****
status : *****
objet : *****
emis: *****

==message==
*****
```

| Nom paramètre            | Description                                             |
|--------------------------|---------------------------------------------------------|
| <code>id</code>          | identifiant du mail                                     |
| <code>date_envoi</code>  | date d'envoi du mail                                    |
| <code>password</code>    | mot de passe éventuel pour protéger le mail             |
| <code>fn_download</code> | code md5 permettant de récupérer les fichiers           |
| <code>status</code>      | statut de confirmation (identique liste mail)           |
| <code>objet</code>       | sujet du mail                                           |
| <code>emis</code>        | "description" : champs input:confirmé:date_confirmation |
| <code>message</code>     | le message lui-même                                     |

`champs input` est soit :

- mailTo
- mailCC
- mailBCC

`confirmé` est soit :

- t (vrai)
- f (faux)

`date_confirmation` est présent si `confirmé` est vrai

Pour récupérer les fichiers il suffit d'interroger :

`modules/mail/template/download.php?filename=mail.zip&root=fn_download`

## 6.6. Envoi d'un mail

Nom du script : ``/modules/mail/api/send-mail.php`

Méthode HTTP d'appel : `POST`

Paramètre du script :

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

| Nom paramètre | Présence    | Format                | Description                                                                               |
|---------------|-------------|-----------------------|-------------------------------------------------------------------------------------------|
| mailTo        | obligatoire | Liste d'adresse email | Destinataires                                                                             |
| mailCC        | facultatif  | Liste d'adresse email | Mail en copie                                                                             |
| mailBCC       | facultatif  | Liste d'adresse email | Mail en copie caché                                                                       |
| objet         | obligatoire | Chaîne                | sujet du mail                                                                             |
| message       | obligatoire | Chaîne                | message                                                                                   |
| password      | facultatif  | Chaîne                | Mot de passe de protection du message                                                     |
| uploadFileN   | facultatif  | File                  | Fichier à attaché (autant que souhaité)<br>uploadFile1 pour le premier,<br>uploadFileN... |

La liste des adresses email est formatée comme suit :

- chaque adresse est séparée par une virgule
- une adresse est soit une adresse unique, soit un groupe
- une adresse unique est écrite de la manière suivante : "description" email
- un groupe est écrit de la manière suivante : nom du groupe (groupe)

Exemple :

"eric" eric@sigmalis.com mailto:eric@babette.com, "pierre" pierre@adullact.org, service comptable (groupe)

Retour :

- Si le mail a été accepté par le système :

```
ok:id_mail
```

- Sinon :

```
erreur:raison de l'erreur
```

## 6.7. Suppression d'un email

Nom du script : `/modules/mail/api/delete-mail.php`

Méthode HTTP d'appel : **GET**

Paramètre du script :

| Nom paramètre | Présence    | Format | Description                                              |
|---------------|-------------|--------|----------------------------------------------------------|
| id            | obligatoire | entier | Identifiant du mail retourné par le script list-mail.php |

Retour :

- OK si le mail a été supprimé
- ERREUR suivie de la raison de l'erreur sinon.