Swissdec-Adapter

Manuel d'utilisation

Historique des versions			
Version 1.0	2013-08-07	Marco Stettler	Version 1.0
Version 1.2	2014-01-30	Christoph Brunner	Recovery Tool & Patch Release 2.0_2
Version 1.3	2015-03-02	Marco Stettler	Patch Release 2.0_4
Version 1.4	2016-06-29	Marco Stettler	Release 2.1_0
Version 1.5	2017-06-19	Marco Stettler	Patch Release 2.1_1
Version 1.6	2020-09-15	Christoph Brunner	Version 3.0_1

Table des matières

1. Introduction 1
1.1. Aperçu Swissdec 1
1.2. Swissdec-Adapter 1
1.3. Interfaces 2
1.3.1. Swissdec 2
1.3.2. sM-Client 2
1.4. Références 3
2. Exigences liées au fonctionnement 4
2.1. Exigences du système 4
2.1.1. Plates-formes supportées 4
2.1.2. Démarrage comme service 4
2.1.3. Accessibilité 4
2.1.4. CPU 4
2.1.5. Mémoire vive (RAM) 5
2.1.6. Disque dur 5
2.1.7. Temps système 5
2.2. Piste de migration 6
3. Installation et configuration 7
3.1. Paquet de livraison 7
3.2. Installation sur Linux 8
3.2.1. Préparation du système 8
3.2.2. Installation des applications 8
3.2.3. Installation du service 9
3.3. Installation sur Windows 10
3.3.1. Préparation du système 10
3.3.2. Installation de l'application 10
3.3.3. Installation du service 10
3.4. Configuration de l'intégration SwissdecAdapter 12
3.4.1. Configuration du système 12
3.4.2. Configuration de l'application 12
3.4.3. Configuration des données 13
3.4.4. Configuration du répertoire 14
3.5. Configuration SwissdecAdapter Receiver 15
3.5.1. Configuration du système 15
3.5.2. Configuration de l'application 15
3.5.3. Configuration de sécurité 16
3.5.4. Connectivité 16
4. Sécurité 18
4.1. Transport Layer (SSL/TLS) 18
4.2. Webservice Security 20
5. Notes sur l'opération 22
5.1. Fenêtre de maintenance 22
5.2. Configuration du logging 23
5.3. Monitoring 23
5.3.1. Monitoring intégré 23
5.4. Console Admin 23
5.4.1. Resources 23
5.5. Test de l'installation 24
5.5.1. Installation 24
5.5.2. Configuration 24
5.5.3. Exécution des tests 25
6. Les problèmes fréquents et leurs solutions 26
A. Annexe 27
A.1. Documents référencés 27
A.2. Glossaire 27
A.3. Comparaison de la configuration 2.x à 3.x 27
A.3.1. 27
A.3.2. 28
B. Exemples 30
B.1. Configuration 30

Liste des illustrations

1.1. Croquis d'installation Swissdec-Adapter 1
1.2. Interface du système de fichiers de l'adaptateur Swissdec sM-client 2
4.1. Aperçu de la sécurité 18
4.2. SSL Handshake 19
4.3. Protocole Handshake SSL avec authentification mutuelle 19
4.4. Transport Layer Security Overview 19
4.5. Webservice Security Overview 20

Liste des tableaux

1.1. Chemins d'accès de l'interface Swissdec-Adapter - sM-Client 3
3.1. swissdecAdapter-delivery-3.0_1.zip 7
3.2. Description de la configuration WinSW 10
3.3. Intégration Swissdec-Adapter configuration de base (integration/conf/application.properties) 12
3.4. Intégration SwissdecAdapter Integration configuration de base (integration/conf/application.properties) .. 12
3.5. Intégration SwissdecAdapter configuration de base 13
3.6. Swissdec-Adapter répertoire 14
3.7. SwissdecAdapter Receiver configuration du système (receiver/conf/application.properties) 15
3.8. SwissdecAdapter Receiver configuration de l'application (receiver/conf/application.properties) 15
3.9. SwissdecAdapter Receiver WS-Security (receiver/conf/application.properties) 16
3.10. SwissdecAdapter Receiver SSL/TLS Security (receiver/conf/application.properties) 16
4.1. Légende de l'image aperçu de la sécurité 18
4.2. Webservice Security Signature 20
4.3. Webservice Security Encryption 20
5.1. Ajustage du log 23
5.2. Configuration du TestTool 24
A.1. Swissdec Adapter Receiver - Version 2.x vs. 3.x 27
A.2. Swissdec Adapter Integration - Version 2.x vs. 3.x 28

Liste des exemples

B.1. Configuration standard réception 30
B.2. Configuration standard intégration 30

1. Introduction

Ce document décrit l'installation, la configuration et le fonctionnement de l'adaptateur *Swissdec*. L'adaptateur Swissdec est une application Java, composé de deux applications web, pour le remplacement des notifications Swissdec concernant les domaines Tax [taxe] (attestations de salaire), TaxAtSource [retenues à la source] (décomptes de l'impôt à la source) ainsi que TaxCrossborder (déclaration pour les frontaliers). Conçu pour être intégré avec le *sM-Client* , il peut cependant aussi être utilisé de manière autonome.

1.1. Aperçu Swissdec

Swissdec est un projet commun à but non lucratif de plusieurs partenaires indépendants et le label de qualité pour l'échange électronique de données entre les entreprises et les assureurs ainsi que les autorités publiques.

En tant que plate-forme d'information centrale pour la normalisation de l'échange électronique de données, Swissdec offre les services suivants :

- Swissdec fournit un savoir-faire pour la normalisation
- sert à l'échange d'informations entre toutes les parties intéressées, telles que les utilisateurs de logiciels, les fabricants d'ERP, les entreprises, les associations, les administrations et les organisations
- surveille la transmission sécurisée des données
- certifie les programmes salariaux testés avec succès.

LA MISSION DE SWISSDEC

Swissdec simplifie l'échange de données entre les entreprises et les partenaires impliqués dans Swissdec sous les aspects suivants:

- Élaboration de normes
- Securité et protection des données pour tous
- Des économies pour tous
- Assurance de la qualité

La plate-forme Swissdec est une plate-forme client-serveur orientée vers les services, avec le distributeur comme intermédiaire. Le schéma suivant donne un bref aperçu de l'architecture du Swissdec:

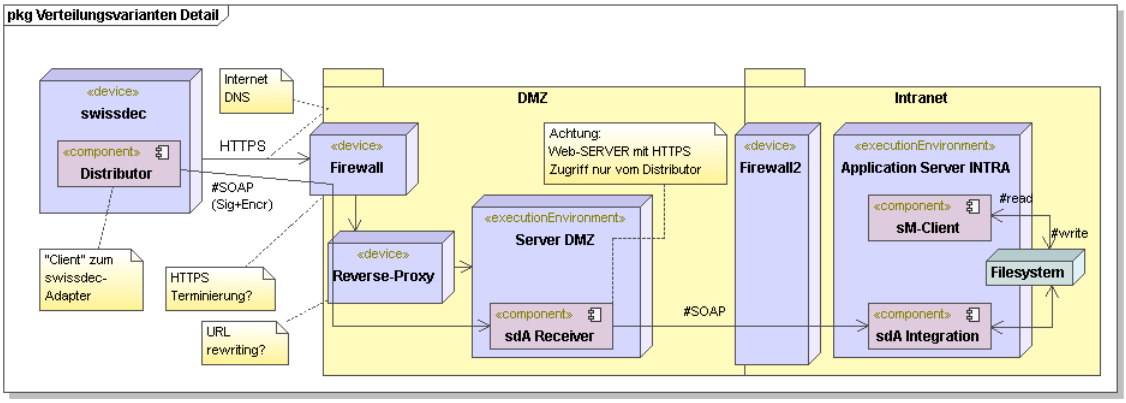


Figure 1.1. Croquis d'installation Swissdec-Adapter

1.2. Swissdec-Adapter

L'adaptateur Swissdec offre les fonctionnalités suivants:

- Réception et envoi de notifications *Swissdec* conformes au protocole
- Transmission sécurisée des données salariales au moyen des protocoles SSL (Secure Sockets Layer) et TLS (Transport Layer Security) ainsi que WS-Security (Web Services Security)
- Vérification de l'intégrité au moyen de la signature WS-Security

La plate-forme Swissdec est synchrone: chaque notification transmise par une société du distributeur est immédiatement traitée et redirigée vers le destinataire final. La société (source des données salariales) voit ainsi la totalité du système comme un ensemble. Si un destinataire final devait ne pas être traité avec la qualité requise, la fiabilité de tout le système serait réduite. Tous les participants doivent donc convenir d'une fiabilité minimale.

Nous désirons de prendre le point de vue du client. Ça veut dire qu'une solution internet s'adaptera aux heures d'ouverture courant. D'un autre côté les aspects économiques d'opération du destinataire (abordabilité) doivent aussi être respectés.

La disponibilité des systèmes doit être vu comme objectif prévu. Ça veut dire que l'importance de la solution augmente et cependant la disponibilité. L'objectif est une solution pragmatique ("lightweight construction" et "best effort"). Plage de temporisation défini:

- 7 jours par semaine à 24 heures
- Heures d'affluence: 6h à 20h

Domaine des valeurs défini:

- Heures d'affluence: Disponibilité des destinataires (m2m) 99.52% aux minimum
- Autres: Disponibilité des destinataires (m2m) 93.00% aux minimum

1.3. Interfaces

1.3.1. Swissdec

L'interface Swissdec a déjà été implémentée dans le produit STEP de la société itServe, installée par plusieurs clients et appliquée par Swissdec. La documentation Swissdec peut être consultée sur le site internet http://www.swissdec.ch de l'association.

1.3.2. sM-Client

Une interface basée sur les fichiers intervient entre l'adaptateur Swissdec et le *sM-Client*. L'adaptateur inscrit les notifications détaillées dans un répertoire-boîte de réception, qui sera interrogé par le *sM-Client*.

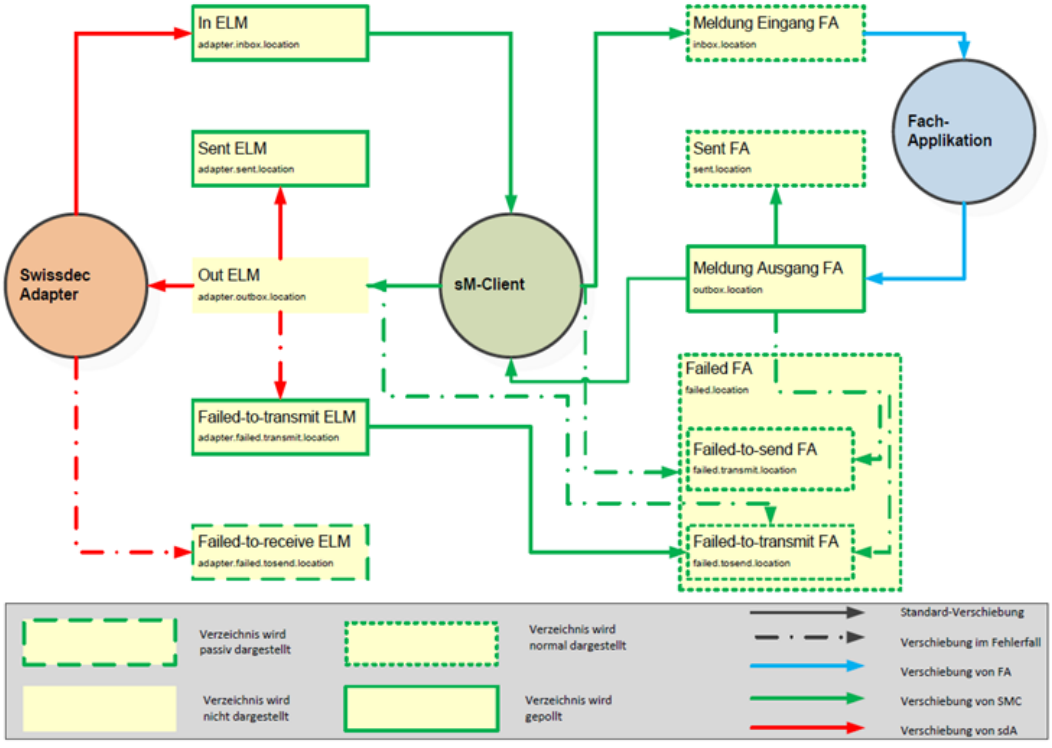


Figure 1.2. Interface du système de fichiers de l'adaptateur Swissdec sM-client

sM-Client (Message-handler-elm.properties)	**PathID**	**Swissdec-Adapter (application.properties)**
Im sM-Client sind diese Verzeichnisse relativ zu base.dir.adapter		sdA braucht absoluten Pfad in dieser Konfigurationsdatei
adapter.inbox.location	PATH_ELM_IN	released.filesystem.parent.xml
adapter.outbox.location	PATH_ELM_OUT	result.filesystem.parent.xml
adapter.sent.location	PATH_ELM_SENT	sent.filesystem.parent.xml
adapter.failed.receive	PATH_ELM_FAILED_TO_RX	failed.filesystem.parent.xml
adapter.failed.transmit.location	PATH_ELM_FAILED_TO_TX	undeliverable.filesystem. parent.xml

Tableau 1.1. Chemins d'accès de l'interface Swissdec-Adapter - sM-Client

Pour la spécification complète de l'interface, veuillez vous référer à la [DETAILSPEZ]

1.4. Références

- [ENDRECREQ] exigences Swissdec – destinataire final
- [DETAILSPEZ] spécification détaillée - adaptateur Swissdec

2. Exigences liées au fonctionnement

L'adaptateur Swissdec se compose de deux applications Java. Les deux applications (réception et intégration) nécessitent un Java Runtime. Le package d'installation contient les applications préconfigurées et peut de ce fait être facilement décompressé.

L'application-réception implémente le protocole Swissdec et doit pouvoir être accessible depuis l'internet via HTTPS. Typiquement, celle-ci est installée dans le *DMZ*.

L'application-intégration assure la communication avec le *sM-Client*. Cette interface est basée sur le système de fichiers. Cela signifie que l'application-intégration et le *sM-Client* ont besoin d'un système de fichiers commun. L'application-intégration peut être installée aussi bien sur le même système que l'application-réception que sur un système dédié ou sur le système du *sM-Client*.

La communication entre les deux applications se produit au moyen des protocoles HTTP et SOAP. Il s'agit de s'assurer qu'une telle connexion puisse être ouverte depuis l'application-réception (client, initie toujours la connexion) vers l'application-intégration.

2.1. Exigences du système

2.1.1. Plates-formes supportées

L'adaptateur Swissdec est une simple application Java. Une version 11 de Java JDK doit être installée. Plates-formes testées et supportées:

- Ubuntu 20.04 LTS
- Windows Server 64bit
- Windows 10

2.1.2. Démarrage comme service

En raison de l'architecture synchrone de Swissdec, l'adaptateur Swissdec doit fonctionner en permanence. Afin de garantir ce fonctionnement permanent, l'adaptateur Swissdec devrait être configuré comme Service (sur Windows) ou comme Script au démarrage (sur Unix).

2.1.3. Accessibilité

L'adaptateur Swissdec doit être accessilble depuis le distributeur selon les exigences Swissdec. Les détails sont spécifiés dans le chapitre connectivité Section 3.5, « Configuration SwissdecAdapter Receiver » dans la partie Receiver.

2.1.4. CPU

Les exigences relatives au CPU sont faibles. L'utilisation de processeurs actuel de 64-bit multi-cœurs est recommandée.

Note

En raison de l'architecture Swissdec, le problème se pose que le distributeur doit attendre tous les destinataires finaux avant d'envoyer la quittance au client. C'est pourquoi Swissdec a spécifié les exigences relatives à la performance suivantes :

- [ENDRECREQ], chapitre 3.10 : Une notification de salaire avec 100 personnes devrait être traitée en moins de 20 secondes.
- [ENDRECREQ], chapitre 3.10 : Une notification de salaire avec 100 personnes doit pouvoir être traitée en une minute.
- [ENDRECREQ], chapitre 3.10 : Une notification de salaire avec 2 000 personnes devrait être traitée en une minute.

2.1.5. Mémoire vive (RAM)

Un espace d'au moins 1 gigaoctet (Go) devrait être réservé pour l'adaptateur Swissdec. Cela permet de recevoir des notifications jusqu'à environ 6 000 salariés. Au cas où des notifications plus grandes devaient être reçues, nous recommandons de réserver un espace de 4 gigaoctets. Cela permet de recevoir des notifications jusqu'à environ 20 000 salariés.

Note

Depuis la version 2.1_0 l'adaptateur Swissdec traite la sécurité basée sur les flux. En théorie, la consommation de RAM devrait être plus efficace.

2.1.6. Disque dur

L'installation nécessite environ 150 mégaoctets (Mo). L'espace nécessaire sur le disque dur pour la durée de fonctionnement varie de manière significative et dépend du nombre et de la taille des notifications reçues. Comme valeur initiale, nous recommandons un espace de 1 gigaoctet sur le disque dur.

2.1.7. Temps système

Il est important que l'heure système du système sur lequel l'adaptateur Swissdec (en particulier le récepteur) fonctionne soit correcte. En effet, un horodatage est généré au niveau de la sécurité des services Web. Si l'heure du distributeur et celle du récepteur final s'écartent, les messages sont rejetés. Nous vous recommandons de mettre à jour l'heure automatiquement en utilisant le NTP (Network Time Protocol).

2.2. Piste de migration

La mise à jour de la version 2.x à la version 3.0 peut être effectuée sans perte de données. Pour ce faire, il faut suivre les étapes suivantes. La procédure est décrite ici en gros. Les détails de l'installation sont décrits dans Section 3.3, « Installation sur Windows » etSection 3.2, « Installation sur Linux ».

1. Mettez une fenêtre de maintenance dans l'ancienne installation afin qu'un message d'erreur correct apparaisse lorsque la migration est effectuée.

2. Arrêt des anciens services de SwissdecAdapter (Integration et Receiver)

3. Sauvegarde de la base de données

   La base de données se trouve dans le dossier derbydb. Le dossier derbydb doit être sauvegardé. Si la nouvelle application démarre correctement la base de données fournie peut être remplacé par le dossier de l'ancienne installation.

4. Nous recommandons de sauvegarder la configuration de l'ancienne installation. La configuration contient les fichiers suivants:

   <SDA_INTEGRATION>/conf/server.xml
   <SDA_INTEGRATION>/conf/swissdecAdapter.properties
   <SDA_RECEIVER>/conf/server.xml
   <SDA_RECEIVER>/conf/swissdecAdapter.properties

   Les différences de configuration entre 2.x et 3.x sont décrit dansSection A.3, « Comparaison de la configuration 2.x à 3.x ».

5. Sauvgarde des certificats:

   Le chemin vers les certificats peut être trouvé dans les deux fichiers de configuration du receveur. Ces fichiers doivent être sauvegardés, parce qu'on en a besoin dans la nouvelle installation.

6. Installation de Java 11 (OpenJDK est recommandé)

7. Déballage, installation et configuration des nouvelles applications SwissdecAdapter selon Section 3.4, « Configuration de l'intégration SwissdecAdapter » et Section 3.5, « Configuration SwissdecAdapter Receiver » .

8. Tester la nouvelle installation avec l'outil de test fourni

9. Arrêter l'intégration de SwissdecAdapter et remplacer la base de données vide fournie par la base de données sauvegardée (Dossier derbydb de l'ancienne installation) et redémarrer l'integration.

10. L'adaptateur Swissdec version 3.0.0 est maintenant installé.

3. Installation et configuration

3.1. Paquet de livraison

Le paquet de livraison, qui est livré sous forme de fichier zip, contient la documentation, les applications et l'outil de test pour tester la fonctionnalité de l'adaptateur Swissdec au niveau local.

Dossier	**Description**
doc	Contient le manuel d'utilisation et des spécifications détaillées
integration	Contien l'application SwissdecAdapter-Integration (swissdecAdapter-integration.jar), la configuration standard (application.properties), la base de données (derbydb) et les fichiers wrapper d'installation Windows Service.
receiver	Contien l'application SwissdecAdapter-Receiver (swissdecAdapter-receiver.jar), la configuration standard (application.properties) et les fichiers wrapper d'installation Windows Service.
swissdecAdapter-testtool.zip	Outil de test avec application de test, données de test et documentation de test

Tableau 3.1. swissdecAdapter-delivery-3.0_1.zip

3.2. Installation sur Linux

L'installation a été testé sur Ubuntu 20.04 LTS, mais peut être installée sur n'importe quelle dérivé de Linux.

Note

Les instructions d'installation partent du principe que l'adaptateur Swissdec est installé dans le répertoire / opt / swissdecAdapter. Bien entendu, il peut également être installé dans un autre répertoire.

L'application est lancée dans les instructions par l'utilisateur système "springboot".

Ces instructions ne prétendent pas de fonctionner sur tous les systèmes, mais visent à donner à un administrateur de système les directives nécessaires pour que l'installation puisse être effectuée de manière indépendante.

3.2.1. Préparation du système

L'adaptateur Swissdec a besoin d'un Java Development Kit (JDK) dans la version 11.

L'installation peut être fait par apt, yum ou manuellement. Il est important que les variables d'environnement soient correctement définies, afin que la commande "java -version" génère la version JDK correspondante.

Il est recommandé de démarrer l'adaptateur Swissdec avec un utilisateur du système.

3.2.2. Installation des applications

Si aucun utilisateur du système n'existe encore, il est recommandé d'en créer un :

```
sudo useradd -r springboot
```

Les dossiers integration et receiver du paquet de livraison Section 3.1, « Paquet de livraison » sont déballés dans le dossier /opt.

Les fichiers "wrapper" peuvent être supprimés, ils ne sont nécessaires que pour l'installation de Windows.

```
unzip swissdecAdapter-delivery-3.0_1.zip
sudo mkdir /opt/swissdecAdapter
sudo cp -r swissdecAdapter-delivery-3.0_1/integration /opt/swissdecAdapter/.
sudo cp -r swissdecAdapter-delivery-3.0_1/receiver /opt/swissdecAdapter/.
sudo rm /opt/swissdecAdapter/*/*wrapper*
```

Avant que les applications puissent être lancées correctement, les autorisations correspondantes doivent être configurées:

```
sudo chown -R springboot:springboot /opt/swissdecAdapter/integration
sudo chown -R springboot:springboot /opt/swissdecAdapter/receiver
sudo chmod +x /opt/swissdecAdapter/integration/swissdecAdapter-integration.jar
sudo chmod +x /opt/swissdecAdapter/integration/swissdecAdapter-receiver.jar
```

Les applications sont maintenant prêtes à être lancées. Il est recommandé de commencer par le répertoire d'installation, car les détails de la configuration de la base de données sont définis de manière relative.

```
cd /opt/swissdecAdapter/integration
java -jar swissdecAdapter-integration.jar
cd /opt/swissdecAdapter/receiver
java -jar swissdecAdapter-receiver.jar
```

3.2.3. Installation du service

3.2.3.1. /etc/init.d Service

L'installation en tant que service fonctionne avec un lien logiciel vers le fichier Jar. Comme la version n'est pas incluse dans le nom du fichier, le service fonctionnera aussi après les futures mises à jour.

```
sudo ln -s /opt/swissdecAdapter/integration/swissdecAdapter-integration.jar
/etc/init.d/swissdecAdapter-integration
sudo update-rc.d swissdecAdapter-integration defaults
sudo ln -s /opt/swissdecAdapter/integration/swissdecAdapter-receiver.jar
/etc/init.d/swissdecAdapter-receiver
sudo update-rc.d swissdecAdapter-receiver defaults
```

3.2.3.2. Installation de systemd

Aux installations modernes de Linux les installations sont souvent effectuées via systemd. Les fichiers suivants sont nécessaires :

/etc/systemd/system/swissdecAdapter-integration.service

```
[Unit]
Description=SwissdecAdapter Integration
After=syslog.target

[Service]
User=springboot
ExecStart=/opt/swissdecAdapter/integration/swissdecAdapter-integration.jar
SuccessExitStatus=143

[Install]
WantedBy=multi-user.target
```

/etc/systemd/system/swissdecAdapter-receiver.service

```
[Unit]
Description=SwissdecAdapter Integration
After=syslog.target

[Service]
User=springboot
ExecStart=/opt/swissdecAdapter/integration/swissdecAdapter-receiver.jar
SuccessExitStatus=143

[Install]
WantedBy=multi-user.target
```

3.3. Installation sur Windows

L'installation a été testée sur Windows 10, mais peut être installée sur n'importe quel système Windows moderne.

Note

Les instructions d'installation supposent que l'adaptateur Swissdec est installé dans le répertoire C:\swissdecAdapter. Bien entendu, il est également possible d'installer l'adaptateur dans un autre répertoire.

Ces instructions ne prétendent pas de fonctionner sur tous les systèmes, mais visent à donner à un administrateur de système les directives nécessaires pour que l'installation puisse être effectuée de manière indépendante.

3.3.1. Préparation du système

L'adaptateur Swissdec a besoin d'un Java Development Kit (JDK) de la version 11.

L'installation peut être effectuée via un Windows Installer ou par installation dans n'importe quel répertoire. Il est avantageux que les variables d'environnement soient correctement définies, de sorte que la commande de la ligne de commande "java -version" génère la version correspondante du JDK.

3.3.2. Installation de l'application

Les dossiers integration et receiver du paquet de livraison Section 3.1, « Paquet de livraison » sont déballés dans le dossier C:\swissdecAdapter .

Les applications sont maintenant prêtes à être lancées. Il est recommandé de démarrer à partir du répertoire d'installation, car les paramètres de configuration de la base de données sont définis de manière relative.

```
cd C:\swissdecAdapter\integration
java -jar swissdecAdapter-integration.jar
cd /opt/swissdecAdapter/receiver
java -jar swissdecAdapter-receiver.jar
```

3.3.3. Installation du service

Pour installer les applications SwissdecAdapter en tant que service, le Windows Service Wrapper (github.com/winsw/winsw [github.com/winsw/winsw]) est inclus. L'exécutable avec la configuration correspondante se trouve dans le répertoire integration resp. receiver.

Pour installer les services, les commandes suivantes doivent être émises dans command prompt (exécutée en tant qu'administrateur):

```
C:\swissdecAdapter\integration\swissdecAdapter-integration-wrapper.exe install
C:\swissdecAdapter\receiver\swissdecAdapter-receiver-wrapper.exe install
```

La configuration du wrapper de service peut être ajustée dans le fichier swissdecAdapter - * - wrapper.xml:

Property	**Description**
id	ID unique du futur service Windows
name	Nom du service
description	Description du service
executable	Exécutable à exécuter. Si la version Java ne doit pas être démarrée à partir de la variable d'environnement, vous pouvez également passer directement à un java.exe à partir d'une version Java installée en alternance.
arguments	Arguments derrière l'executable.
logpath	Ici, les logs sont écrits.

Property	Description
log mode	La valeur par défaut est passée à "none". Le log de l'application est contrôlé par Spring Boot (application.properties)

Tableau 3.2. Description de la configuration WinSW

3.4. Configuration de l'intégration SwissdecAdapter

La configuration de l'intégration de SwissdecAdapter peut être trouvée dans le fichier application.properties dans le dossier integration/conf.

3.4.1. Configuration du système

Option	Valeur standard	Description
server.port	9090	Port TCP sur lequel l'application est lancée. Doit être configuré dans la partie destinataire.
derby.system.home	derbydb	Accès à la base de données
housekeeping.days	180	Combien de jours la declaration doit être sauvegardée.
monitoring.enabled	false	La page de monitoring est-elle lancée ? (Accessible sous http://[HOST]:[PORT]/api/monitoring)
monitoring.user	admin	Nom d'utilisateur pour la page de monitoring
monitoring.pass	admin	Mot de passe pour l'accès à la page monitoring

Tableau 3.3. Intégration Swissdec-Adapter configuration de base (integration/conf/application.properties)

3.4.2. Configuration de l'application

Option	Valeur standard	Description
institution.canton	BE	Spécification technique pour laquelle des institutions les messages impôt à la source sont reçus (p.ex. „BE“, „ZH“, „GE“, ...). DOIT être configuré correctement. Sinon les messages sont rejetés.
await.result	false	Faut-il s'attendre à des résultats comptables de l'impôt à la source? N'est utilisé que si une intégration au backend est disponible.
processing.default.hours	48	Après combien d'heures peut-on s'attendre à un éventuel résultat comptable de l'impôt à la source.
testcase.auto.quittance	false	Empêche l'intégration back-end lors de la réception d'une déclaration de test. (seulement actif si await.result=true)
write.original.xml	false	Écrit le fichier original complet dans le fichier ZIP lorsque le fractionnement est actif.
commune.splitting	false	Activation du fractionnement des communes pour le décompte de l'impôt à la source. Si le fractionnement des communes est actif, await.result est implicitement "false", car les communes ne peuvent pas fournir de réponses.
la.splitting	false	Activation du fractionnement des certificats de salaire par personne reçue.

Option	Valeur standard	Description
map.tas.toV5	false	Les messages impôt à la source sont transférés de ELM v4.0 à ELM v5.0, si la condition de map.tas.fromPeriod est accompli.
map.tas.fromPeriod	2021-01	Si map.tas.toV5 est activé tous les messages impôt à la source qui sont plus récents ou égals à la période configurée sont transférés. La comparaison est fait avec CurrentMonth de la declaration de salaire. (SalaryDeclaration/ Staff/ Person/ TaxAtSourceSalaries/ TaxAtSourceSalary/ CurrentMonth)

Tableau 3.4. Intégration SwissdecAdapter Integration configuration de base (integration/conf/ application.properties)

3.4.3. Configuration des données

Option	Valeur standard	Description
released.filesystem.parent.xml	data/received	Chemin d'accès vers l'entrée de la notification (PATH_ELM_IN). Demande de données de chemin d'accès absolu. Doit correspondre avec le sM-client. Exemple pour Windows: C:/sdA/data/received
failed.filesystem.parent.xml	data/failed	Chemin d'accès vers "failed to receive" (PATH_ELM_FAILED_TO_RX). Demande de données de chemin d'accès absolu. Doit correspondre avec le sM-client. Exemple pour Windows: C:/sdA/data/failed
result.filesystem.parent.xml	data/result	Chemin d'accès vers la sortie de la notification (PATH_ELM_OUT). Demande de données de chemin d'accès absolu. Doit correspondre avec le sM-client. Exemple pour Windows: C:/sdA/data/result
sent.filesystem.parent.xml	data/sent	Chemin d'accès vers « sent » (PATH_ELM_SENT). Demande de données de chemin d'accès absolu. Doit correspondre avec le sM-client. Exemple pour Windows: C:/sdA/data/sent
undeliverable.filesystem.parent.xml	data/undeliverable	Chemin d'accès vers "failed to receive" (PATH_ELM_FAILED_TO_RX). Demande de données de chemin d'accès absolu. Doit correspondre avec le sM-client.

Option	Valeur standard	Description
		Exemple pour Windows: C:/sdA/data/undeliverable

Tableau 3.5. Intégration SwissdecAdapter configuration de base

3.4.4. Configuration du répertoire

Le tableau suivant donne un aperçu des dossiers dans SwissdecAdapter qui sont pertinent pour la transmission des déclarations.

PathID	Description
PATH_ELM_IN	• sont au format standard Swissdec : XML (validés par rapport à un schéma) • ont comme nom de fichier : ["Tax"|"TaxAtSource"|"Tax5"|"TaxAtSource5"|"TaxCrossborder5"]_DeclarationId_ResultIdentifier_["MIXD"|"SPLT"].zip • les notifications utilisées seront déplacées/supprimées par le sM-client
PATH_ELM_OUT	• Des réponses ne seront créées que pour les décomptes de l'impôt à la source (résultat des décomptes de l'impôt à la source) • sont au format standard Swissdec : XML (validés par rapport à un schéma); zippés • ont comme nom de fichier : DeclarationId.zip • les notifications utilisées seront déplacées par l'adaptateur Swissdec • les notifications non utilisées seront déplacées par l'adaptateur Swissdec
PATH_ELM_SENT	• Résultats des décomptes de l'impôt à la source envoyés • sont au format standard Swissdec : XML (validés par rapport à un schéma); zippés • ont comme nom de fichier : DeclarationId_ResultIdentifier.zip • seront lus par le sM-client afin de pouvoir clôturer le processus
PATH_ELM_FAILED_TO_RX	• Attestation de salaire et décompte de l'impôt à la source, erreur à la réception • les réponses de l'adaptateur Swissdec seront également inscrites • non disponible dans chaque cas
PATH_ELM_FAILED_TO_TX	• Résultats des décomptes de l'impôt à la source, erreur à l'envoi

Tableau 3.6. Swissdec-Adapter répertoire

3.5. Configuration SwissdecAdapter Receiver

La configuration du SwissdecAdapter Receiver se trouve dans le fichier application.properties dans le dossier receiver/conf.

3.5.1. Configuration du système

Option	Valeur standard	Description
server.port	8080	TCP Port sur lequel l'application est démarré.
integration.service.host	localhost	L'intégration de SwissdecAdapter fonctionne sur ce host. (Peut être aussi une adresse IP, si un DNS n'est pas disponible.)
integration.service.port	9090	TCP Port sur lequel l'application fonctionne.
integration.service.protocol	http://	Quel est le protocole utilisé pour accéder à l'intégration SwissdecAdapter.

Tableau 3.7. SwissdecAdapter Receiver configuration du système (receiver/conf/application.properties)

3.5.2. Configuration de l'application

Option	Valeur standard	Description
institution.canton	BE	Spécification technique pour laquelle des institutions les messages impôt à la source sont reçus (p.ex. „BE“, „ZH“, „GE“, ...). DOIT être configuré correctement. Sinon les messages sont rejetés.
tax.accept.ex	false	Définit si les certificats de salaire pour les personnes résidant à l'étranger sont acceptés. Veuillez consulter votre département spécialisé pour savoir si cette option est souhaitée. Si oui veuillez nous contacter, s.v.p. (JIRA oder E-Mail), parce que ceci doit être configuré auprès du distributeur.
institution.name	KSTV Bern	Nom de l'administration fiscale.
elm.tac.enabled	true	L'installation est prête à recevoir des certificats de salaire des frontaliers.
elm.tax.enabled	true	L'installation est prête à recevoir des certificats de salaire avec ELM v5.0.
elm.tas.enabled	true	L'installation est prête à recevoir des messages d'impôt à la source avec ELM v5.0.
elm4.tax.enabled	true	L'installation est prête à recevoir des certificats de salaire avec ELM v4.0.
elm4.tas.enabled	true	L'installation est prête à recevoir des messages d'impôt à la source avec ELM v4.0.

Tableau 3.8. SwissdecAdapter Receiver configuration de l'application (receiver/conf/application.properties)

3.5.3. Configuration de sécurité

La sécurité est traitée séparément dans le chapitre sécurité Chapitre 4, *Sécurité* .

3.5.3.1. WS-Security

Le certificat pour la sécurité WS est sauvegardé dans un Java Keystore (*.jks). La configuration est fait comme suivant :

Option	Standardwert	Beschreibung
wss4j.org.apache.ws.security.crypto.provider	org.apache.wss4j.common.crypto.Merlin	Security Provider
wss4j.org.apache.ws.security.crypto.merlin.file	security/TestConsumerAll.jks	Chemin au Java Keystore
wss4j.org.apache.ws.security.crypto.merlin.keystore.type	jks	Keystore Typ
wss4j.org.apache.ws.security.crypto.merlin.keystore.password	default	Mot de passe du Java Keystore pour décrypter
wss4j.org.apache.ws.security.crypto.merlin.keystore.alias	1	nom alias
wss4j.privateKeyPassword	default	Mot de passe du Private Key

Tableau 3.9. SwissdecAdapter Receiver WS-Security (receiver/conf/application.properties)

3.5.3.2. SSL/TLS Security

La configuration du Transport Layer Security (TLS) peut être repris directement du Swissdec-Adapter. Il est conseillé de faire le HTTPS Traffic sur un autre système (p.ex.: Apache, WAF, etc.).

Option	Standardwert	Beschreibung
server.ssl.key-store	security/TestConsumerAll.jks	Pfad zum Keystore
server.ssl.key-store-type	JKS	Typ des Keystore (JKS oder PKCS)
server.ssl.key-store-password	default	Passwort zum Java Keystore
server.ssl.key-alias	1	Alias Name
server.ssl.key-password	default	Passwort zum Private Key

Tableau 3.10. SwissdecAdapter Receiver SSL/TLS Security (receiver/conf/application.properties)

3.5.4. Connectivité

L'adaptateur Swissdec démarre une extrémité d'un service web, qui doit être accessible depuis l'internet. Un chemin d'accès depuis l'internet vers l'adaptateur Swissdec doit également être créé. En outre, les éléments suivants doivent être configurés :

- URL: Pour pouvoir être accessible depuis l'internet, l'adaptateur Swissdec a besoin de cinq URL. L'association Swissdec stipule qu'il faut attribuer une adresse unique pour chaque domaine (impôt à la source / TaxAtSource [ELM v4.0 & v5.0], impôt / Tax [ELM v4.0 & v5.0], TaxCrossborder). Plus d'informations à ce sujet à la Section 3.5.4.1, « Reverse Proxy ».
- Protocoles SSL/TLS: Afin de garantir la connexion, l'association Swissdec stipule des protocoles SSL/TLS avec une authentification des clients : outre le HTTPS « normal » un certificat est demandé aux clients puis vérifié. Plus d'informations à ce sujet à laSection 3.5.4.2, « SSL/TLS ».
- Pare-feu: L'adaptateur Swissdec n'ouvre aucune connexion dans l'internet. C'est pourquoi seul le chemin d'accès détaillé doit être configuré. La connexion peut être limitée au distributeur. Plus d'informations à ce sujet à laSection 3.5.4.3, « Firewall ».

3.5.4.1. Reverse Proxy

Au cas où un proxy inverse est en place, les URLs suivantes doivent être accessibles depuis l'extérieur:

```
http://<HOST>:<PORT>/webservice/tax/SalaryDeclarationConsumerService
```

```
http://<HOST>:<PORT>/webservice/tas/SalaryDeclarationConsumerService
```

```
http://<HOST>:<PORT>/webservice/tax/SalaryDeclarationConsumerService20200220
```

```
http://<HOST>:<PORT>/webservice/tas/SalaryDeclarationConsumerService20200220
```

```
http://<HOST>:<PORT>/webservice/tac/SalaryDeclarationConsumerService20200220
```

3.5.4.2. SSL/TLS

L'association Swissdec stipule que, pour le destinataire final, outre le HTTPS « normal » le client (distributeur) doit s'authentifier. Ceci afin de protéger le destinataire final. En principe, le destinataire final (utilisateur) décide comment et où planifier le protocole SSL/TLS avec authentification mutuelle. Les solutions suivantes peuvent alors être envisagées :

- La planification survient sur le serveur d'entrée ou le pare-feu.
- La planification survient sur le receveur de l'adaptateur Swissdec. Vous trouverez la configuration concernée en annexe.

Les certificats pour les deux solutions sont consultables via le support *sM-Client* (https://jira.ctp-consulting.com/browse/SMCSUPPORT).

3.5.4.3. Firewall

Des connexions détaillées doivent être autorisées sur l'adaptateur Swissdec. L'adaptateur Swissdec Receiver démarre par défaut sur le port 8080.

Note

L'adaptateur Swissdec étant utilisé par un client (distributeur), la connexion détaillée peut être limitée à son adresse IP. Pour le distributeur effectif, l'adresse IP est:

- 194.11.148.11

Le distributeur de test, les adresses sont:

- 193.247.121.163
- 193.247.102.165

4. Sécurité

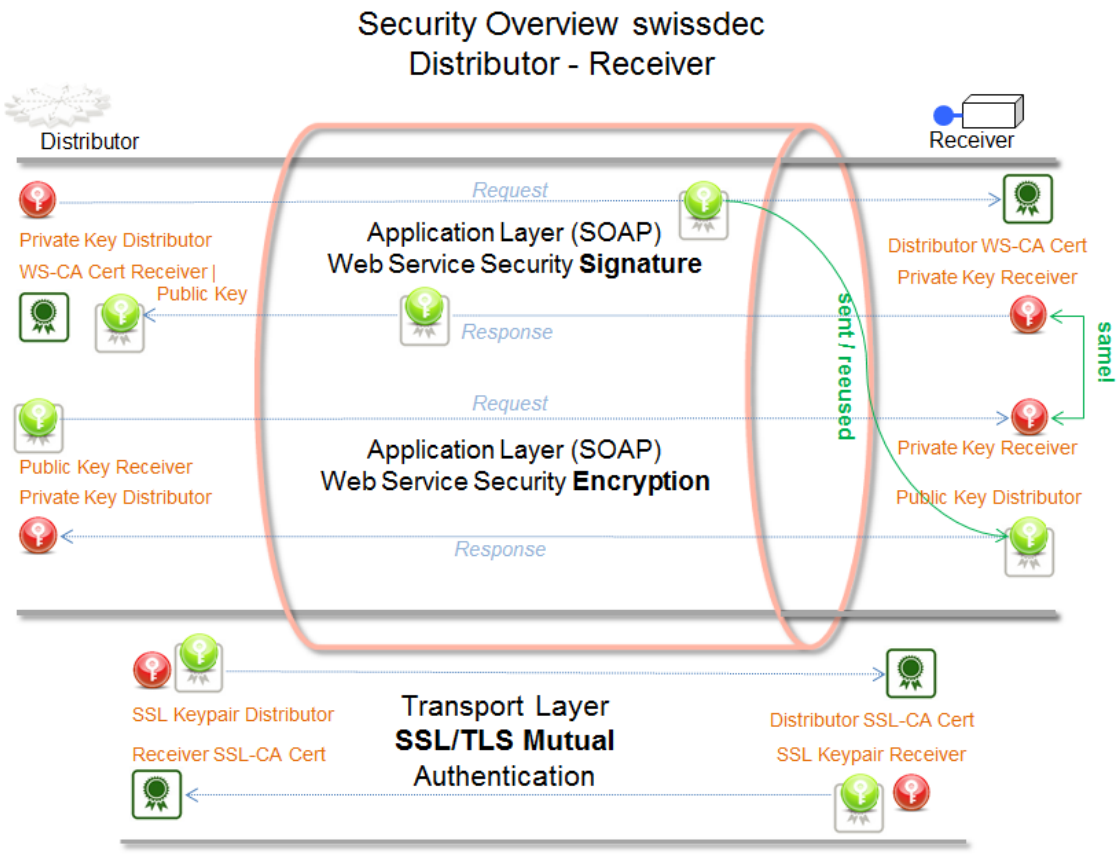


Figure 4.1. Aperçu de la sécurité

	Clef privée (Private Key)
	Certificat avec clé publique (Public Key). Dans ce qui suit, les "clés publiques" (public Key) seront identifiées avec le certificat qui contient la clé publique.
	Certificat CA

Tableau 4.1. Légende de l'image aperçu de la sécurité

4.1. Transport Layer (SSL/TLS)

Le client crée une connexion vers le serveur. Habituellement, le serveur s'authentifie d'abord par rapport au client avec un certificat. Ensuite, soit le client envoie au serveur un nombre aléatoire secret encodé avec la clé publique du serveur, soit les deux parties prévoient un secret commun avec la méthode de l'échange de clés Diffie-Hellman. Une clé cryptographique sera alors déduite du secret. Cette clé sera par la suite utilisée pour encoder toutes les informations relatives à la connexion à l'aide d'une procédure d'encodage symétrique et pour garantir l'intégrité et l'authenticité des informations au moyen d'un code d'authentification de message.

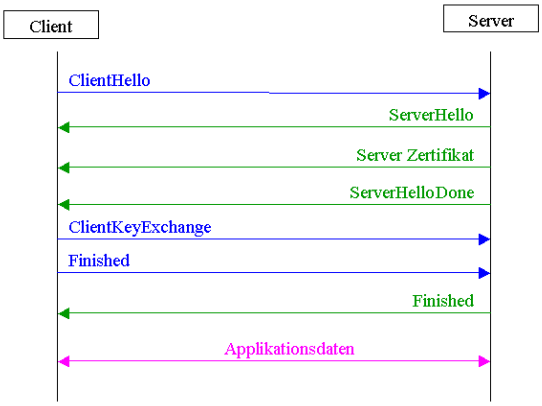


Figure 4.2. SSL Handshake

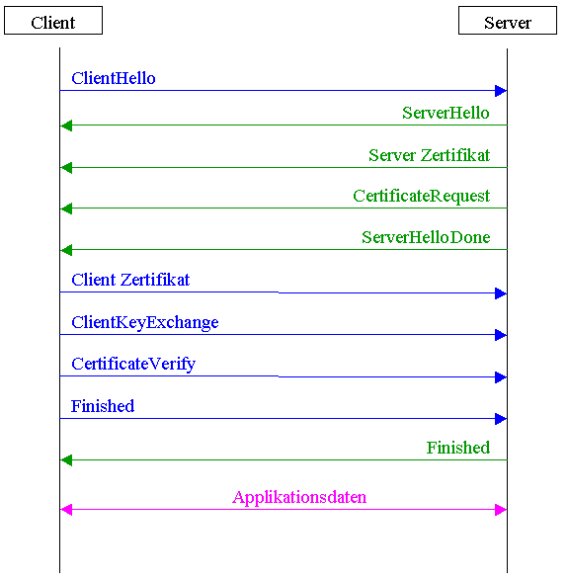


Figure 4.3. Protocole Handshake SSL avec authentification mutuelle

Dans le cadre de l'authentification mutuelle, le serveur exige également du client qu'il s'authentifie avec un certificat. Cette exigence est stipulée dans les normes Swissdec afin de garantir que toutes les parties se font mutuellement confiance.

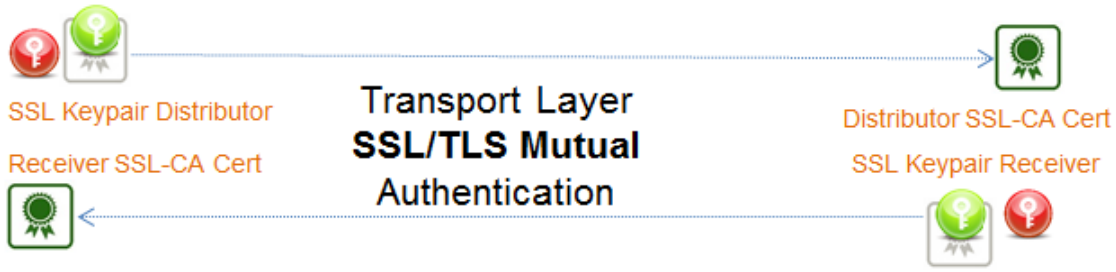


Figure 4.4. Transport Layer Security Overview

Pour simplifier, il est évident que les deux côtés (client/distributeur Swissdec, serveur/adaptateur Swissdec) devront utiliser aussi bien une paire de clés (clé privée avec la clé publique qui y correspond) pour s'identifier qu'un certificat CA pour la vérification.

La configuration des protocoles SSL/TLS (Tableau 3.10, « SwissdecAdapter Receiver SSL/TLS Security (receiver/conf/application.properties) ») pour l'adaptateur Swissdec se trouve dans le fichier "application.properties". Si une planification devait être effectuée sur l'adaptateur, un exemple de configuration se trouve dans ce même fichier.

Note

Keystore (avec la paire de clés) et le fichier Truststore (avec le certificat CA du distributeur Swissdec) seront livrés dans un seul fichier pour simplifier. Ainsi, les deux données sont identiques!

Les certificats effectifs peuvent être commandés via le support sM-client (https://smcsupport.atos-solutions.ch/). En outre, le nom de l'hôte (entrée DNS [système de noms de domaine]) doit être communiqué et être accessible depuis l'internet avec l'installation de l'adaptateur Swissdec.

4.2. Webservice Security

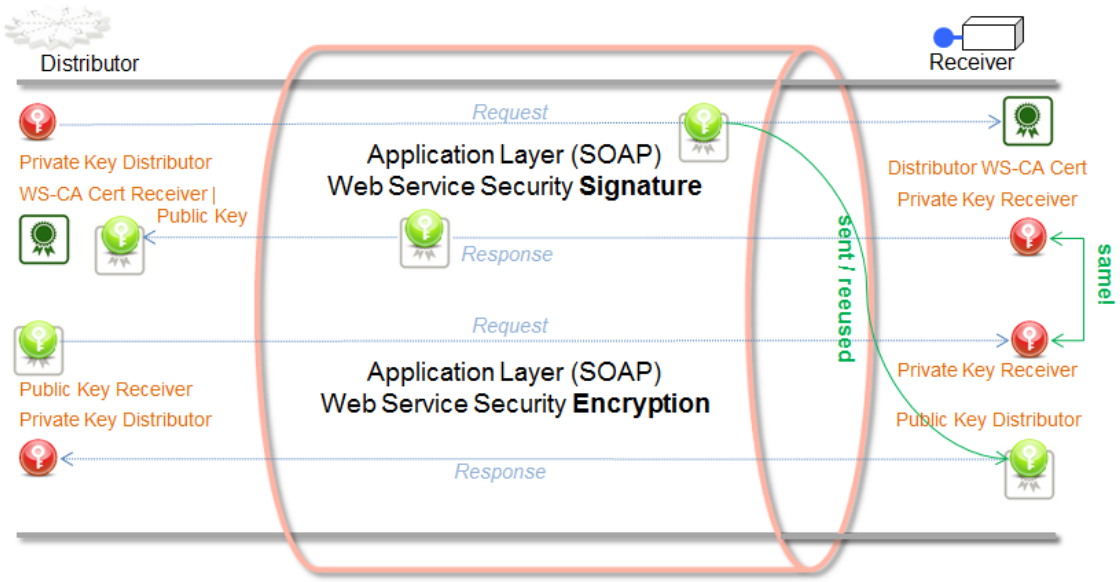


Figure 4.5. Webservice Security Overview

Signature

Request	Le client (distributeur Swissdec) signe la requête avec sa clé privée (Private Key) et envoie sa clé publique (Public Key) avec la requête. Sur la page du serveur (adaptateur Swissdec), la clé publique envoyée avec la requête est vérifiée par rapport au certificat CA Swissdec, de même que la signature. Au niveau du serveur, seul le certificat CA Swissdec est nécessaire pour vérifier la signature.
Response	L'adaptateur Swissdec signe la réponse avec sa clé privée. Le distributeur Swissdec doit connaître sa clé publique, ou le certificat CA.

Tableau 4.2. Webservice Security Signature

Pour le processus de signature du protocole Webservice Security, le certificat CA du client ainsi que sa propre paire de clés (clé privée et clé publique) seront également nécessaires du côté du serveur.

Cryptage

Request	Le client (distributeur Swissdec) encode la requête avec la clé publique (Public Key) du destinataire. Sur la page du serveur (adaptateur Swissdec), la requête est déchiffrée avec la clé privée correspondante.
Response	L'adaptateur Swissdec encode la réponse avec la clé publique du distributeur Swissdec qui a été envoyée avec la requête pour la signature du protocole Webservice Security. Il est ainsi garanti que seul le propriétaire de la clé privée correspondante peut déchiffrer la réponse.

Tableau 4.3. Webservice Security Encryption

Pour le processus d'encryptage du protocole Webservice Security, seule la clé privée du destinataire, qui est identique à celle nécessaire pour la signature Webservice Security, est nécessaire du côté serveur.

Configuration

L'illustration 3.2 montre que du côté du serveur (adaptateur Swissdec), deux choses sont nécessaires pour la sécurité du WebService :

- Certificat CA Swissdec
- Paire de clés (clé privée et clé publique) du destinataire

Comme le certificat CA doit seulement être connu, il n'est pas configuré explicitement. Il suffit qu'il existe dans *Keystore* configuré. Toutefois, la clé privée du destinataire final doit être explicitement adressée dans le *Keystore*. Les options de configuration suivantes (Tableau 3.9, « SwissdecAdapter Receiver WS-Security (receiver/conf/application.properties) ») (application.properties) contrôlent le comportement de l'adaptateur Swissdec pour le Webservice Security.

Note

Les certificats effectifs peuvent être commandés via le support *sM-Client* Support (http://jira.ctp-consulting.com).

Avertissement

Dès que les certificats Webservice Security sont installés pour la production, aucune transmission provenant aussi bien des "refapps" (reference application) que des outils de test ne sera acceptée. Cela est dû au fait que les signatures des notifications ne peuvent plus être vérifiées, le distributeur effectif ayant utilisé des certificats d'une autre autorité de certification (CA).

5. Notes sur l'opération

5.1. Fenêtre de maintenance

Car les transmissions sur la plateforme Swissdec sont synchrones, les clients voient quelles destinataires sont atteignables ou non. Il est possible d'annoncer des interruptions de fonctionnement prévues au moyen de la fenêtre de maintenance. On devrait planifier ceci à préférence entre 20:00 et 06:00 les jours de travail ou le week-end. Comme ça les clients sont informés avec un message au lieu d'une erreur. Pour enregistrer une fenêtre de maintenance, procédez comme suivant:

1. Créez un fichier XML « PlannedMaintenance ». Vous trouverez un modèle (PlannedMaintenance.xml) dans le répertoire integration/templates .

2. Sauvegardez ce fichier dans le répertoire « maintenance.filesystem.parent.xml » de l'application-intégration de l'adaptateur Swissdec. Le nom du fichier n'a aucune importance; l'extension du fichier doit être ".xml".

3. La fenêtre de maintenance sera signalée au prochain Ping du distributeur. Accessibilité du destinataire est testé par le distributeur cycliquement (toutes les 30 minutes). Pour ceci l'opération "PingConsumer" est éxécutée. L'accessibilité est confirmée par le destinataire. Dans ce processus le destinataire peut configurer des fenêtres de maintenance dans sa réponse.

4. Dès que la fenêtre de maintenance est passée, le fichier XML "PlannedMaintenance" est automatiquement supprimé.

Note

- Le nom du fichier n'a aucune importance, l'extension du fichier doit obligatoirement être ".xml".
- Les pages "Start" et "End" dans le fichier XML "PlannedMaintenance" sont du type xs:dateTime. Cela signifie que les fuseaux horaires, ainsi que les heures d'été et d'hiver, doivent y être contenus.
  - 2013-08-31T13:40:47.0Z : Le Z désigne l'UTC
  - 2014-01-31T13:40:47.0+01:00 : UTC +1 heure, heure de l'Europe Centrale (heure d'hiver)
  - 2013-08-31T13:40:47.0+02:00: UTC +2 heures, heure de l'Europe Centrale (heure d'été)
- Les notifications seront transmises jusqu'à la comptabilité des salaires. Veuillez sélectionner le texte approprié dans les 3 langues (allemand, français et italien).
- Plusieurs fichiers XML « PlannedMaintenance » peuvent être classés dans le répertoire "maintenance.filesystem.parent.xml". L'adaptateur Swissdec notifiera toujours le fichier suivant au distributeur ; et au cours du fonctionnement, le fichier sera automatiquement supprimé ou le fichier suivant sera notifié.

Exemple d'un fichier XML "PlannedMaintenance" :

```
<PlannedMaintenance xmlns="http://www.itserve.ch/step/core"
                xmlns:ct="http://www.itserve.ch/step/coreTypes">
   <ct:Start>2050-01-01T00:00:00.0+01:00</ct:Start>
   <ct:End>2050-01-01T01:00:00.0+01:00</ct:End>
   <ct:Messages>
      <ct:Language>de</ct:Language>
      <ct:Value>Test Wartungsfenster sdA</ct:Value>
   </ct:Messages>
   <ct:Messages>
      <ct:Language>fr</ct:Language>
      <ct:Value>Fenêtre de maintenance sdA</ct:Value>
   </ct:Messages>
   <ct:Messages>
      <ct:Language>it</ct:Language>
      <ct:Value>Finestra di manutenzione sdA</ct:Value>
   </ct:Messages>
</PlannedMaintenance>
```

5.2. Configuration du logging

La configuration du logging se trouve pour chaque application dans le fichier application.properties.

Option	Default	Description
logging.level.ch.itserve.step	INFO	Log Level de tous les messages dans le package ch.itserve.step
logging.file	log/swissdecAdapter-receiver.log	Les logs sont écrit dans ce fichier.
logging.file.max-history	1	Indique de combien de jours les logs restent présent dans le fichier avant d'être supprimé automatiquement.

Tableau 5.1. Ajustage du log

Pour plus d'informations, veuillez consulter le site suivant: https://docs.spring.io/spring-boot/docs/2.1.13.RELEASE/reference/html/boot-features-logging.html

5.3. Monitoring

L'adaptateur Swissdec peut être contrôlé comme tout autre processus ou service standard. Si l'adaptateur Swissdec a été enregistré en tant que service Windows, il est possible d'utiliser la « console Windows Gestion de l'ordinateur » pour vérifier ou modifier l'état du service.

5.3.1. Monitoring intégré

La surveillance intégrée (interface REST) est décrite dans le chapitre console d'administration.

5.4. Console Admin

Pour l'authentification à l'interface REST, les options suivantes sont disponible dans la configuration :

```
monitoring.user = admin
monitoring.pass = admin
```

L'interface RESTful est atteignable avec l'URL suivant :

```
http://<INTEGRATION_HOST>:<INTEGRATION_PORT>/integration/api/
```

Lors de la récupération, veuillez vous assurer que le HTTP-Request Header "Accept: application/json" est défini. L'authentification est garantie par l'utilisation de HTTP-BASIC. Exemple request:

```
GET http://localhost:8280/integration/api/
Accept: application/json
Username: admin
```

Vous trouvez plus d'informations concernant l'authentification sous Wikipedia [https://en.wikipedia.org/wiki/Basic_access_authentication]

5.4.1. Resources

5.4.1.1. Statistiques

Aperçu:

```
http://<INTEGRATION_HOST>:<INTEGRATION_PORT>/integration/api/statistics
```

5.4.1.2. Monitoring

Atteignable par l'URL:

```
http://<INTEGRATION_HOST>:<INTEGRATION_PORT>/integration/api/monitoring
```

Exemple de résultat du Monitoring intégré:

```
{

   "connectivity": {
      "lastpingDateTime": "20.12.2016 09:37:44",
      "lastpingTimestamp": "1482223064619"
   },
   "database": {
      "path": "derbydb",
      "version": "3.0_1 (Build 24685)"
   },
   "system": {
      "java": "Oracle Corporation 1.8.0_131 (/usr/lib/jvm/jdk1.8.0_131/jre)",
      "arch": "Linux 4.4.0-79-generic (amd64)",
      "appVersion": "2.1_1 (Build 22586)",
   },
   "settings": {
      ...
   },
   "systemproperties": {
      ...
   }

}
```

5.5. Test de l'installation

Pour pouvoir vérifier l'installation, Swissdec met à disposition un outil de test. Il s'agit d'un simple service web-client, avec lequel les attestations de salaires et les décomptes de l'impôt à la source peuvent être envoyées au format Swissdec ELM v5.0.

5.5.1. Installation

L'outil de test est livré dans un fichier ZIP et doit seulement être décompressé. Une installation Java sera nécessaire pour l'exécution.

5.5.2. Configuration

Avant de pouvoir procéder au test, les fichiers

- config/tas20130514.xml
- config/tas20130514.xml
- config/tas20200220.xml
- config/tax20200220.xml
- config/tac20200220.xml

doivent être adaptés. Il s'agit de fichiers Properties au format XML.

Option	**Description**
endpointURL	L'adresse de l'adaptateur Swissdec à tester.
wssEncryptionCert	Le certificat utilisé pour l'encodage de la notification (clé publique de l'adaptateur Swissdec).
sigrootcert	Le certificat utilisé pour vérifier la signature de la réponse (clé publique de l'adaptateur Swissdec).
sslcert	Certificat avec la clé publique pour la clé privée SSL
sslkey	Clé privée SSL pour l'authentification client

Option	Description
sslpartnercacert	Certificat CA pour la vérification du certificat de serveur SSL/TLS.

Tableau 5.2. Configuration du TestTool

Pour plus d'informations concernant la sécurité, veuillez consulter la Chapitre 4, *Sécurité*.

Note

La configuration fournie correspond à la configuration par défaut de l'adaptateur Swissdec.

L'option « sslpartnercacert » ne sera nécessaire qu'avec un HTTPS activé.

Les options « sslcert » et « sslkey » ne seront nécessaires qu'avec une authentification mutuelle activée.

5.5.3. Exécution des tests

Pour procéder à un test, le script batch ou shell correspondant sera exécuté. Une documentation des cas de test est fourni avec le TestTool.

6. Les problèmes fréquents et leurs solutions

Notification

```
Caused by: org.apache.ws.security.components.crypto.CredentialException: Failed to load credentials.
   Inner Exception: [Keystore was tampered with, or password was incorrect]
```

Mot de passe *Keystore* erroné dans Swissdec-Adapter Receiver application.properties (Property 'org.apache.ws.security.crypto.merlin.keystore.password').

Notification

```
Caused by: java.io.FileNotFoundException: abTst_server.jks (No such file or directory)
```

Le *Keystore* référencé dans Swissdec-Adapter Receiver application.properties n'existe pas (Property 'org.apache.ws.security.crypto.merlin.file')

Notification

```
Caused by: java.security.UnrecoverableKeyException: Cannot recover key
```

Mot de passe de la clé privée est erroné Swissdec-Adapter Receiver application.properties (Property 'wss4j.privateKeyPassword')

Notification

```
Caused by: javax.xml.ws.soap.SOAPFaultException: Marshalling Error: Connection refused
 at org.apache.cxf.jaxws.JaxWsClientProxy.invoke(JaxWsClientProxy.java:156)
 at $Proxy205.saveSalaryDeclaration(Unknown Source)
```

L'application Receiver n'a aucune connexion avec l'intégration. Veuillez contrôler la configuration suivante dans Receiver application.properties:

```
integration.service.protocol=http://
integration.service.host=localhost
integration.service.port=9090
```

Assurez-vous que la configuration est adaptée à votre infrastructure et qu'une connexion HTTP peut être ouverte sur l'URL configurée à partir de l'application Receiver.

A. Annexe

A.1. Documents référencés

[ENDRECREQ] *Directives relatives à la transmission de données salariales.*. Endreceiver Requirements. Swissdec. Version 5 (20200220).

[DETAILSPEZ] *Swissdec-Adapter*. Detailspezifikation. itServe AG. V02.01.

A.2. Glossaire

DMZ	Un réseau informatique avec des possibilités d'accès, contrôlées en matière de sécurité, sur le serveur correspondant
Keystore	Un Keystore est une banque de données des clés et des certificats dans un fichier.
NTP	Standard pour la synchronisation des montres des systèmes informatiques.
sM-Client	Le sM-client offre une fonctionnalité pour l'envoi et la réception de notifications dans différents formats
STEP	Produit de la société itServe AG, destinataire *Swissdec*.
Swissdec	Label de qualité pour les systèmes de comptabilité des salaires. Standards XML pour la procédure unifiée de communication des salaires (ELM).

A.3. Comparaison de la configuration 2.x à 3.x

Pour simplifier la migration les différences de la configuration des versions 2.x et 3.x sont listées ci-dessous :

2.x	**3.x**	**Differenz**
	server.port	Dans 2.x la configuration est dans conf/server.xml
	server.ssl.key-store-type	Dans 2.x la configuration est dans conf/server.xml
	server.ssl.key-store	Dans 2.x la configuration est dans conf/server.xml
	server.ssl.key-store-password	Dans 2.x la configuration est dans conf/server.xml
	server.ssl.key-alias	Dans 2.x la configuration est dans conf/server.xml
	server.ssl.key-password	Dans 2.x la configuration est dans conf/server.xml
userAgent.institutionName	institution.name	Configuration remplacée
institution.tax.id	institution.canton	Configuration regroupée
institution.tas.id	institution.canton	Configuration regroupée
	elm.tac.enabled	Nouveau feature
	elm.tas.enabled	Nouveau feature
	elm.tax.enabled	Nouveau feature
	elm4.tas.enabled	Nouveau feature
	elm4.tax.enabled	Nouveau feature
tax.accept.ex	tax.accept.ex	Pas de changement
salarydeclaration.service.protocol	integration.service.protocol	Configuration remplacée
salarydeclaration.service.host	integration.service.host	Configuration remplacée

2.x	3.x	Differenz
salarydeclaration.service.port	integration.service.port	Configuration remplacée
salarydeclaration.service.context		N'est plus nécessaire
salarydeclaration.service.name		N'est plus nécessaire
	wss4j.org.apache.ws.security. crypto.provider	Nouvelle configuration
org.apache.ws.security. crypto.merlin.file	wss4j.org.apache.ws.security. crypto.merlin.file	Configuration remplacée
org.apache.ws.security. crypto.merlin.keystore.type	wss4j.org.apache.ws.security. crypto.merlin.keystore.type	Configuration remplacée
org.apache.ws.security. crypto.merlin.keystore.password	wss4j.org.apache.ws.security. crypto.merlin.keystore.password	Configuration remplacée
org.apache.ws.security. crypto.merlin.alias.password		N'est plus nécessaire
org.apache.ws.security. crypto.merlin.keystore.alias	wss4j.org.apache.ws.security. crypto.merlin.keystore.alias	Configuration remplacée
wss4j.privateKeyPassword	wss4j.privateKeyPassword	Pas de changement
wss4j.signatureUser=1		N'est plus nécessaire
monitoring.enabled		N'est plus nécessaire
monitoring.user		N'est plus nécessaire
monitoring.pass		N'est plus nécessaire
	logging.level.ch.itserve.step	Nouveau feature
	logging.file	Nouveau feature
	logging.file.max-history	Nouveau feature

Tableau A.1. Swissdec Adapter Receiver - Version 2.x vs. 3.x

2.x	3.x	Différence
	server.port	Dans 2.x la configuration est dans conf/server.xml
institution.tax.id	institution.canton	Configuration regroupée
institution.tas.id	institution.canton	Configuration regroupée
released.filesystem.parent.xml	released.filesystem.parent.xml	Pas de changement
failed.filesystem.parent.xml	failed.filesystem.parent.xml	Pas de changement
result.filesystem.parent.xml	result.filesystem.parent.xml	Pas de changement
sent.filesystem.parent.xml	sent.filesystem.parent.xml	Pas de changement
undeliverable.filesystem.parent.xml	undeliverable.filesystem.parent.xml	Pas de changement
maintenance.filesystem.parent.xml	maintenance.filesystem.parent.xml	Pas de changement
await.result	await.result	Pas de changement
commune.splitting	commune.splitting	Pas de changement
la.splitting	la.splitting	Pas de changement
testcase.auto.quittance	testcase.auto.quittance	Pas de changement
write.original.xml	write.original.xml	Pas de changement
	map.tas.toV5	Nouveau feature dès 3.x
	map.tas.fromPeriod	Nouveau feature dès 3.x
	derby.system.home	Dans 2.x la configuration est dans System Properties
processing.default.hours	processing.default.hours	Pas de changement

2.x	**3.x**	**Différence**
housekeeping.days	housekeeping.days	Pas de changement
monitoring.enabled	monitoring.enabled	Pas de changement
monitoring.user	monitoring.user	Pas de changement
monitoring.pass	monitoring.pass	Pas de changement
	logging.level.ch.itserve.step	Nouveau feature
	logging.file	Nouveau feature
	logging.file.max-history	Nouveau feature

Tableau A.2. Swissdec Adapter Integration - Version 2.x vs. 3.x

B. Exemples

B.1. Configuration

```
                #
# Copyright 1996-2020 itServe AG. All rights reserved.
#
# This software is the proprietary information of itServe AG
# Bern Switzerland. Use is subject to license terms.
#

# System Settings
server.port=8080

# SwissdecAdapter Receiver Settings
institution.name=KSTV Bern
institution.canton=BE

elm.tac.enabled=true
elm.tas.enabled=true
elm.tax.enabled=true

elm4.tas.enabled=true
elm4.tax.enabled=true

tax.accept.ex=false

# SSL-Security with JKS
#server.ssl.key-store=classpath:security/TestConsumerAll.jks
#server.ssl.key-store-type=JKS
#server.ssl.key-store-password=default
#server.ssl.key-alias=1
#server.ssl.key-password=default

# SSL-Security with PKCS12
#server.ssl.key-store=classpath:keystore/baeldung.p12
#server.ssl.key-store-type=PKCS12
#server.ssl.key-store-password=password
#server.ssl.key-alias=1

# WS-Security
wss4j.org.apache.ws.security.crypto.provider=org.apache.wss4j.common.crypto.Merlin
wss4j.org.apache.ws.security.crypto.merlin.file=security/TestConsumerAll.jks
wss4j.org.apache.ws.security.crypto.merlin.keystore.type=jks
wss4j.org.apache.ws.security.crypto.merlin.keystore.password=default
wss4j.org.apache.ws.security.crypto.merlin.keystore.alias=1
wss4j.privateKeyPassword=default

# Remote Services
integration.service.protocol=http://
integration.service.host=localhost
integration.service.port=9090

# LOG Settings
logging.level.ch.itserve.step=INFO
logging.file=log/swissdecAdapter-receiver.log
logging.file.max-history=1
```

Exemple B.1. Configuration standard réception

```
                #
# Copyright 1996-2020 itServe AG. All rights reserved.
#
# This software is the proprietary information of itServe AG
# Bern Switzerland. Use is subject to license terms.
#

# System Settings
server.port=9090

# SwissdecAdapter Integration Settings
institution.canton=BE
```

```
released.filesystem.parent.xml=data/received
failed.filesystem.parent.xml=data/failed
result.filesystem.parent.xml=data/result
sent.filesystem.parent.xml=data/sent
undeliverable.filesystem.parent.xml=data/undeliverable
maintenance.filesystem.parent.xml=data/maintenance

await.result=false
commune.splitting=false
la.splitting=false
testcase.auto.quittance=false
write.original.xml=false
map.tas.toV5=false
map.tas.fromPeriod=2021-01

derby.system.home=derbydb

processing.default.hours=48
housekeeping.days=180

monitoring.enabled=false
monitoring.user=admin
monitoring.pass=admin

# LOG Settings
logging.level.ch.itserve.step=INFO
logging.file=log/swissdecAdapter-integration.log
logging.file.max-history=
```

Exemple B.2. Configuration standard intégration