SwissdecAdapter

Manuel d'utilisation

Historique des versions			
Version 1.0.0	2013-08-07	Marco Stettler	Release 1.0
Version 2.0.0	2014-01-30	Christoph Brunner	Release 2.0
Version 2.1.0	2016-06-29	Marco Stettler	Release 2.1_0
Version 3.0.0	2020-09-15	Christoph Brunner	Release 3.0_0
Version 3.0.4	2021-03-11	Christoph Brunner	Release 3.0_4
Version 3.1.0	2023-08-22	Christoph Brunner	Release 3.1_0

Table des matières

1. Concernant le release actuel 1
1.1. Apérçu du release 1
1.2. Chemin de migration 2
2. Introduction SwissdecAdapter 3
2.1. Aperçu Swissdec 3
2.2. Swissdec-Adapter 3
2.3. Interfaces 4
2.4. Références 5
3. Exigences opérationnelles 6
3.1. Exigences du système 6
4. Installation 8
4.1. Paquet de livraison 8
4.2. Installation sur Linux 8
4.3. Installation sur Windows 11
5. Configuration 13
5.1. Receiver 13
5.2. Intégration 16
6. Sécurité 20
6.1. Transport Layer TLS 21
6.2. Webservice Security 22
7. Fonctionnement 24
7.1. Configuration du logging 24
7.2. Fenêtre de maintenance 24
7.3. Monitoring 25
7.4. Console d'administration 25
A. Annexe 27
A.1. Apache Reverse Proxy 27
A.2. Testtool 28
A.3. Tableau unicode 29
A.4. Documents référencés 29
A.5. Glossaire 29

1. Concernant le release actuel

Ce chapitre décrit les nouveautés et le chemin de migration de la version actuelle.

1.1. Apérçu du release

En résumé, les principaux changements du SwissdecAdapter 3.1.0

- Update à ELM 5.1 avec l'implémentation du nouveau domaine Swissdec "Unemployment Certificate" (Attestation de chômage).
- Optimisation du HousekeepingJob pour éviter l'exception OutOfMemory
- Les transmissions de participation de collaborateur sont sauvegardées séparément des certificats de salaire
- Les doublons sont identifiés dans Syslog avec le DeclarationID du message original, de façon que le support des clients peut être amélioré
- Validation des réponses, si await.result=true. Les fautes de validation apparaissent dans le log. L'entreprise reçoit un "Processing" (Comportement si le résultat n'est pas encore disponible).
- Upgrade à JDK 17 (Version LTS actuel de JDK)
- Upgrade des bibliothèques et frameworks.

1.2. Chemin de migration

Avec les pas suivants la migration peut être effectuée dans quelques minutes.

Warnung

La fenêtre de maintenance doit être planifiée à l'avance, parce que le distributeur Swissdec ne vérifie que tous les 30 minutes si une fenêtre de maintenance est définie.

Si rien d'autre est décrit, les pas suivants doivent être effectués sur les deux applications SwissdecAdapter.

1. **Arrêter les services**
2. **Sauvegarder la configuration**

   Avant l'update, il est impérativement recommandé de sauvegarder les données suivantes dans un classeur séparé.

   a. Configuration

      Les fichiers application.properties des deux applications

   b. Intégration: sauvegarder la base de données (Classeur "derbydb")

   c. Receiver: sauvegarder les certificats (Dans le classeur "security")
3. **Installer Java 17 sur le système et contrôler la configuration du système**

   Linux: sudo apt install openjdk-17-jdk

   Windows: Installation JDK 17 (p.ex.: Adoptium [https://adoptium.net/])

   Dans le terminal du système, la commande "java -version" doit afficher la version de Java installée. Sinon, la configuration du service doit être adaptée de manière que le service soit lancé avec Java 17.
4. **Adapter la configuration**

   Receiver: application.properties

   - Renommer la clef "logging.file" à "logging.file.name"

   Intégration: application.properties

   - Renommer la clef "logging.file" à "logging.file.name"
5. ### Warnung

   Vérifier la valuer "logging.file.max-history". Si la valeur est 1, le LOG est sauvegardé pour seulement un jour. Il est recommandé de mettre la valeur à 60.
6. **Démarrer les services**

2. Introduction SwissdecAdapter

Ce document décrit l'installation, la migration, la configuration et le fonctionnement du *Swissdec* Adapter. Le SwissdecAdapter est une application Java qui contient deux applications web pour échanger des messages Swissdec dans les domaines Tax (certificats de salaire), TaxAtSource (décomptes impôt à la source), TaxCrossborder (messages frontaliers) et UnemploymentCertificate (attestation de chômage). Il a été développé pour être intégré dans le *sM-Client*, mais il peut aussi être utilisé de manière autonome.

2.1. 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 standardisation de l'échange électronique de données, Swissdec offre les services suivants :

- Swissdec fournit un savoir-faire pour la standardisation
- 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
- <vpara>surveille la transmission sécurisée des données</vpara>
- certifie les programmes salariaux testés avec succès.

<vemphasis>LA MISSION DE SWISSDEC</vemphasis>

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

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

La plateforme Swissdec est une plateforme client-serveur orientée services avec le distributeur comme intermédiaire. L'image suivante montre un aperçu de l'architecture Swissdec:

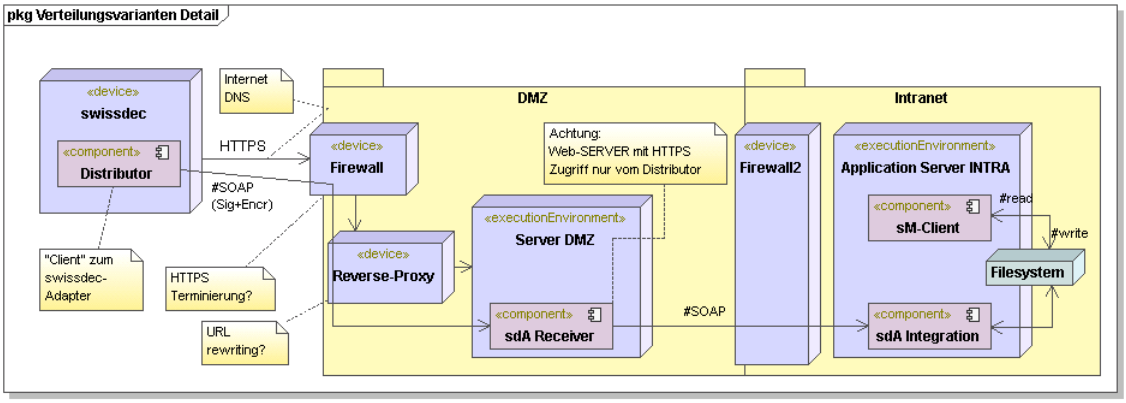


Abbildung 2.1. Esquisse d'installation Swissdec-Adapter

2.2. Swissdec-Adapter

Le Swissdec-Adapter offre les fonctionnalités suivantes:

- Réception et envoi des message *Swissdec* conforme au protocol.
- Transport sécurisé des données de salaire avec SSL/TLS et WS-Security
- Vérification de l'intégrité avec signature de WS-Security

La plateforme Swissdec est synchrone. Cela signifie que chaque déclaration d'une entreprise est immédiatement traitée par le distributeur et transmise aux destinataires finaux. L'entreprise (source des données salariales) vit donc l'ensemble du système comme une unité. Si un destinataire final n'est pas exploité avec la qualité requise, ce récepteur diminue la fiabilité de l'ensemble du système. Tous les participants doivent donc se mettre d'accord sur une fiabilité minimale.

Nous nous efforçons d'adopter le point de vue du client, c'est-à-dire qu'une solution internet ne s'adaptera pas aux heures de bureau habituelles. D'autre part, lors de l'exploitation d'un récepteur final, les aspects économiques (capacité de financement) doivent également être pris en compte.

La disponibilité des systèmes doit être comprise comme une valeur visée dans le futur. C'est-à-dire l'importance de la solution augmente et avec ça sa disponibilité. Le but est une solution pragmatique ("lightweight construction" et "best effort"). Horaires définis:

- 7 jours sur 7, 24 heures sur 24
- Heures pointes: 6 h à 20 h

Gamme des valeurs définie:

- Heures pointes: La disponibilité des destinataires finaux (m2m) doit être de 99,52% au minimum.
- Durant les heures creuses, la disponibilité des destinataires finaux (m2m) doit être de 93,00% au minimum.

2.3. Interfaces

2.3.1. Swissdec

L'interface Swissdec a été déjà implémenté dans le produit STEP d'itServe et a été installé chez plusieurs clients. Le logiciel a été certifié par Swissdec. La documentation Swissdec est disponible sur le site [http://www.swissdec.ch] de l'association.

2.3.2. sM-Client

Entre le Swissdec-Adapter et le *sM-Client* une interface basée sur des fichiers est utilisée. Le sdAdapter écrit les messages entrants dans un répertoire inbox. Celui est lu régulièrement par le *sM-Client*.

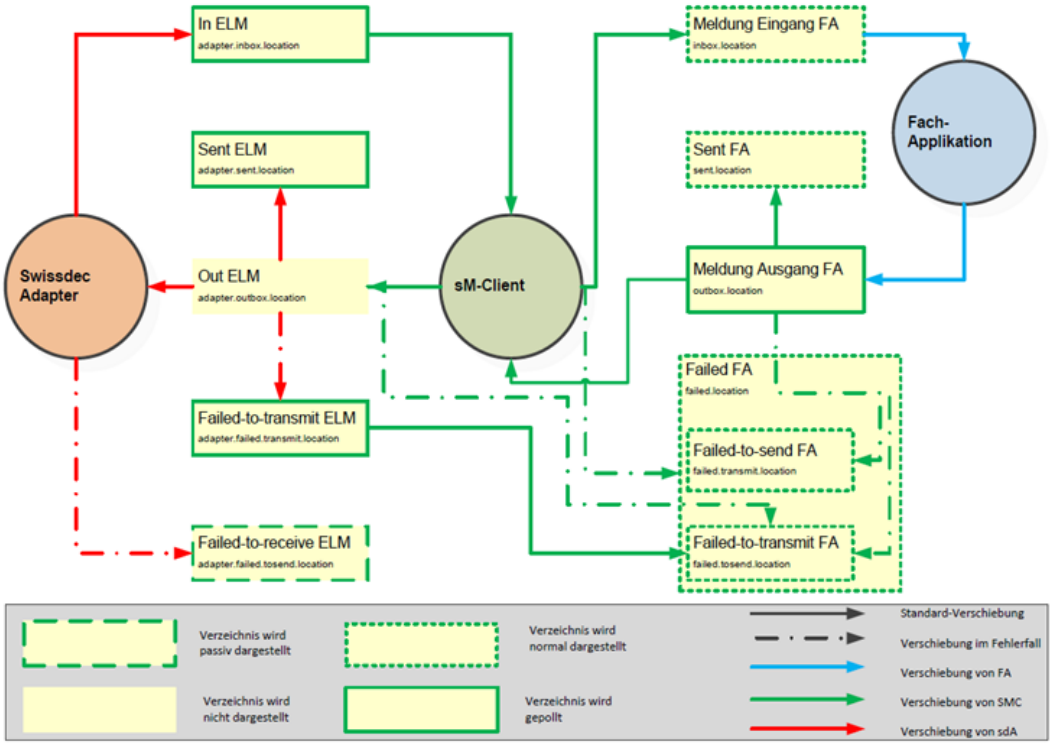


Abbildung 2.2. Interface du système de fichiers Swissdec-Adapter - sM-Client

sM-Client (Message-handler-elm.properties)	**PathID**	**Swissdec-Adapter (application.properties)**
Dans sM-Client ces classeurs sont relatifs à base.dir.adapter		Dans ce fichier de configuration, le sdAdapter a besoin du chemin absolut
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

Tabelle 2.1. Chemins de l'interface Swissdec-Adapter - sM-Client

Vous trouverez les spécifications complètes de l'interface dans le document suivant [DETAILSPEZ]

2.4. Références

- [ENDRECREQ] Exigences pour destinataires Swissdec
- [DETAILSPEZ] Spécification détaillée Swissdec-Adapter

3. Exigences opérationnelles

L'adaptateur Swissdec se compose de deux applications Java. Les deux applications (récepteur et intégration) nécessitent un Java Runtime. Le paquet d'installation contient les applications préconfigurées et peut donc être facilement décompacté.

L'application du récepteur implémente le protocole Swissdec et doit être accessible depuis Internet via HTTPS. Elle est typiquement installée dans le *DMZ*.

L'application d'intégration assure la communication avec le *sM-Client* . Cette interface est basée sur un système de fichiers. C'est-à-dire que l'application d'intégration et le *sM-Client* ont besoin d'un système de fichier commun. L'application d'intégration peut être installée sur le même système que le receveur, sur un système dédié ou sur le système du *sM-Client*.

La communication entre les deux applications se fait par HTTP/SOAP. Il doit être assuré qu'une telle connexion peut être établie par l'application réceptrice (client, qui initie toujours la connexion) vers l'application d'intégration.

3.1. Exigences du système

3.1.1. Plateformes soutenues

Le Swissdec-Adapter est une application Java pure. Un Java JDK en version 17 est préalable. OpenJDK est Es wird eine installiertes Java JDK in der Version 17 vorausgesetzt. OpenJDK est recommandé. Plateformes testées et prises en charge :

- Ubuntu 22.04 LTS
- Windows Server 64bit
- Windows 10 / Windows 11

3.1.2. Démarrer comme service

En raison de l'architecture synchrone Swissdec, l'adaptateur Swissdec doit fonctionner en permanence. Pour garantir cela l'adaptateur Swissdec doit être configuré comme un service (sous Windows) ou un script de démarrage (sous Unix).

3.1.3. Accessibilité

Le SwissdecAdapter doit être accessible selon les directives Swissdec. Les détails concernant ceci se trouvent dans le chapitre connectivité???.

3.1.4. CPU

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

Remarques

L'architecture Swissdec pose le problème que le distributeur doit attendre tous les destinataires finaux avant que la quittance ne soit envoyée au client. Swissdec a donc spécifié les exigences de performance suivantes :

- [ENDRECREQ], chapitre 3.10: Une déclaration de salaires avec 100 personnes devrait être traitée en moins de 20 secondes.
- [ENDRECREQ], chapitre 3.10: Une déclaration de salaires avec 100 personnes doit être traitée en moins de 1 minute.
- [ENDRECREQ], chapitre 3.10: Une déclaration de salaires avec 2000 personnes devrait être traitée en moins de 1 minute.

3.1.5. Mémoire vive

Il faut réserver au moins 1 gigaoctet pour l'adaptateur Swissdec. Cela permet de recevoir des déclarations jusqu'à environ 6'000 collaborateurs. Si des déclarations plus grandes doivent être reçues, nous recommandons de réserver 4 gigaoctets. Cela permet de recevoir des messages jusqu'à environ 20'000 collaborateurs.

Remarque

Depuis la version 2.1_0 le Swissdec-Adapter traite la sécurité du service web par sécurité sur la base d'un stream. La consommation de mémoire est ainsi nettement plus efficace.

3.1.6. Espace mémoire

L'installation nécessite environ 150 mégaoctets. L'espace disque nécessaire au moment de l'exécution varie considérablement et dépend du nombre et de la taille des messages reçus. Comme valeur de départ 1 gigaoctet d'espace libre sur le disque dur est recommandée.

3.1.7. Le temps du système

Il est important que l'heure système du système sur lequel l'adaptateur Swissdec est installé (notamment receveur) est correct. Ceci parce qu'un timestamp est généré au niveau de Web Service Security. Si l'heure du distributeur et celle du receveur final divergent, les messages sont refusés. Il est recommandé d'ajuster l'heure automatiquement avec *NTP (Network Time Protocol)*.

4. Installation

4.1. Paquet de livraison

Le paquet de livraison, livré sous forme de fichier zip, contient la documentation, les applications et l'outil de test pour tester la fonctionnalité du SwissdecAdapter localement.

Registre	Description
doc	Contient le manuel et la spécification détaillée.
integration	Contient l'application SwissdecAdapter-Integration (swissdecAdapter-integration.jar), la configuration standard (integration.properties), la base de données (derbydb) et les fichiers Windows Service-Installations Wrapper.
receiver	Contient l'application SwissdecAdapter-Receiver (swissdecAdapter-receiver.jar), la configuration standard (receiver.properties) et les fichiers Windows Service-Installations Wrapper.
swissdecAdapter-testtool.zip	Testtool avec application de test, données de test et documentation de test

Tabelle 4.1. swissdecAdapter-delivery-3.1_x.zip

4.2. Installation sur Linux

L'installation a été testé sur Ubuntu 22.04 LTS, mais peut être installée sur n'importe quelle dérivée de Linux.

Warnung

Les instructions d'installation partent du principe que Swissdec Adapter 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.

4.2.1. Préparation du système

Le SwissdecAdapter a besoin d'un Java Development Kit (JDK) dans la version 17.

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 SwissdecAdapter avec un utilisateur du système.

4.2.2. Installation des application

S'il n'y a pas encore d'utilisateur système, il est recommandé d'en créer un:

```
sudo useradd -r springboot
```

Les classeurs integration et receiver du paquet de livraison (Abschnitt 4.1, „Paquet de livraison“) sont décompacté dans le classeur /opt.

Les fichiers "wrapper" peuvent être supprimés. Ils sont seulement utilisés pour l'installation sur Windows.

```
unzip swissdecAdapter-delivery-3.1_0.zip
sudo mkdir /opt/swissdecAdapter
sudo cp -r swissdecAdapter-delivery-3.1_0/integration /opt/swissdecAdapter/.
sudo cp -r swissdecAdapter-delivery-3.1_0/receiver /opt/swissdecAdapter/.
sudo rm /opt/swissdecAdapter/*/*wrapper*
```

Avant que les applications peuvent être démarrés correctement, tous les authorisations correspondant doivent être configurés:

```
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
```

Maintenant les applications sont prêtes à démarrer. Il est recommandé de les démarrer dans le répertoire de l'installation, parce que les données de configuration concernant la base de données sont mis de façon relative.

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

4.2.3. Installation du service

4.2.3.1. /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
```

4.2.3.2. Installation de systemd

Aux installations modernes de Linux les installations sont souvent effectuées via systemd. Voici les fichiers pour le faire :

/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
```

4.3. Installation sur Windows

L'installation a été testée sur Windows 10 et 11, mais peut être installé sur chaque système Windows moderne.

Warnung

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

4.3.1. Préparation du système

Le SwissdecAdapter a besoin d'un Java Development Kit (JDK) dans la version 17.

L'installation peut être effectuée par Windows Installer ou par installation dans n'importe quel classeur. Il est préférable que les variables d'environnement soient correctement définies pour que la commande "java -version" affiche la version JDK correspondante.

4.3.2. Installation de l'application

Les classeurs integration et receiver du paquet de livraison (Abschnitt 4.1, „Paquet de livraison“) sont décompacté dans le classeur C:\swissdecAdapter.

Les applications sont prêtes à démarrer. Il est recommandé de les démarrer dans le répertoire de l'installation, parce que les données de configuration concernant la base de données sont mis de façon relative.

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

4.3.3. Installation du service

Pour installer les applications SwissdecAdapter en tant que service, le Windows Service Wrapper est fourni (github.com/winsw/winsw [https://github.com/winsw/winsw]). Le fichier executable avec la configuration correspondante se trouve dans le répertoire integration resp. receiver.

Windows Service Wrapper a besoin du .NET Framework

Pour installer les services, les commandes suivants doivent être exécutées dans Command Prompt (exécuter comme administrateur) :

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

La configuration pour le Service Wrapper peut être adaptée dans le fichier swissdecAdapter-*-wrapper.xml :

Property	**Description**
id	ID unique du futur service Windows
name	Nom d'affichage du service
description	Description du service
executable	Exécutable à exécuter. Si on ne veut pas démarrer le Java définie dans la configuration, il est également possible de changer ici à un java.exe d'une version alternative.
arguments	Arguments après l'executable.
logpath	Les logs sont écrit ici.
log mode	La valeur par défaut est commutée sur "none", il s'agit du log du service wrapper. Le log du SwissdecAdapter est dirigé par Spring Boot (application.properties)

Tabelle 4.2. Description de la configuration WinSW

5. Configuration

5.1. Receiver

5.1.1. Options de configuration du receiver

Dans cette section, les propriétés sont énumérées avec leurs valeurs par défaut et une brève description de l'influence qu'elles peuvent avoir sur le comportement.

5.1.1.1. SwissdecAdapter comportement

Les properties ont une influence sur le comportement de l'application SwissdecAdapter

- **institution.name=KSTV Bern**

  Nom en toutes lettres de l'administration fiscale (voir aussi :Abschnitt A.3, „Tableau unicode“)

- **institution.canton=BE**

  Abréviation du canton

- **elm.tac.enabled=true**

  Le SwissdecAdapter peut recevoir des messages TaxCrossborder (frontaliers)

- **elm.tas.enabled=true**

  Le SwissdecAdapter peut recevoir des messages TaxAtSource (impôt à la source)

- **elm.tax.enabled=true**

  Le SwissdecAdapter peut recevoir des messages Tax (certificat de salaire, participation de collaborateur et attestation de rentes)

- **elm.tuc.enabled=true**

  Le SwissdecAdapter peut recevoir des messages Unemployment Certificate (attestation de chomâge)

- **elm4.tas.enabled=true**

  Le SwissdecAdapter peut recevoir des messages TaxAtSource (impôt à la source) (ELM version 4.0)

- **elm4.tax.enabled=true**

  Le SwissdecAdapter peut recevoir des messages Tax (certificat de salaire, participation de collaborateur et attestation de rentes) (ELM Version 4.0)

- **tax.accept.ex=false**

  Le SwissdecAdapter peut recevoir des certificats de salaire des personnes avec résidence à l'étranger.

5.1.1.2. Configuration du système

Les properties suivants servent à la configuration générale du système

- **server.port=8080**

  L'application SwissdecAdapter-Receiver démarre sur ce port

- **integration.service.protocol=http://**

  Définit par quel protocol SwissdecAdapter-Integration est atteignable

- **integration.service.host=localhost**

Sur cet host est situé l'intégration SwissdecAdapter-Integration

- **integration.service.port=9090**

  L'intégration SwissdecAdapter-Integration fonctionne sur ce port

- **logging.level.ch.itserve.step=INFO**

  Paramétrage du LOG Level (Possibilités : DEBUG, INFO, WARN, ERROR)

- **logging.file=log/swissdecAdapter-receiver.log**

  LOG File standard (est en rotation quotidienne)

- **logging.file.max.history=60**

  Après combien de jours les vieux fichiers log sont supprimés

5.1.1.3. Configuration WS Security (voir également:Kapitel 6, *Sécurité*)

Ces propriétés permettent de contrôler la configuration de Webservice Security

- **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 d'accès au keystore dans lequel le certificat WS est stocké

- **wss4j.org.apache.ws.security.crypto.merlin.keystore.type=jks**

  Type de keystore

- **wss4j.org.apache.ws.security.crypto.merlin.keystore.password=default**

  Mot de passe pour ouvrir le keystore

- **wss4j.org.apache.ws.security.crypto.merlin.keystore.alias=1**

  Alias sous lequel le certificat est sauvegardé dans le keystore

- **wss4j.privateKeyPassword=default**

  Mot de passe pour le Private Key

5.1.1.4. Configuration TLS Security (voir également:Kapitel 6, *Sécurité*)

Ces properties permettent de configurer la sécurité TLS. Ces paramètres sont facultatifs, car la terminaison TLS se fait souvent sur un système en amont (par ex. proxy, passerelle, WAF ou autre).

L'expérience a montré que la terminaison TLS directement dans le SwissdecAdapter Receiver ne fonctionne pas toujours. Si la terminaison TLS est effectuée sur le système sur lequel le SwissdecAdapter Receiver est installé, il est également possible de le faire avec un serveur Apache.

- **server.ssl.key-store=security/TestConsumerAll.jks**

  Chemin d'accès aux certificats TLS fournis dans un *JKS*

- **server.ssl.key-store-type=JKS**

  Type de keystore

- **server.ssl.key-store-password=default**

  Mot de passe du Keystore

- **server.ssl.key-alias=1**

  nom alias du certificat au sein du keystore

- **server.ssl.key-password=default**

  Mot de passe du Private Key

5.1.2. Configuration de référence

```
                #
# Copyright 1996-2024 itServe AG. All rights reserved.
#
# This software is the proprietary information of itServe AG
# Bern Switzerland. Use is subject to license terms.
#


#
# Copyright 1996-2023 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
elm.tuc.enabled=true

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

tax.accept.ex=false

# TLS-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

# TLS-Security with PKCS12
#server.ssl.key-store=classpath:keystore/TestConsumerAll.p12
#server.ssl.key-store-type=PKCS12
#server.ssl.key-store-password=default
#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.name=log/swissdecAdapter-receiver.log
logging.file.max-history=60
```

Exemple 5.1. receiver.properties

5.2. Intégration

5.2.1. Options de configuration pour Integration

5.2.1.1. Comportement du SwissdecAdapter

Ces properties ont une influence au comportement du SwissdecAdapter

- **institution.canton=BE**

  Abbréviation du canton

- **await.result=false**

  Impôt à la source: Si cette valeur est mis à true, SwissdecApapter attend un fichier de réponse qui est retourné à l'entreprise au moment d'un appel GetResult.

- **commune.splitting=false**

  Impôt à la source: Le message impôt à la source est fractionné et écrit dans le fichier zip par Municipality-ID (commune).

- **testcase.auto.quittance=false**

  Si await.true=true, les messages qui sont marqués avec cas de test sont traités directement par le SwissdecAdapter.

- **map.tas.toV5=false**

  Impôt à la source: Les messages ELMv4 sont "mappés" en messages ELMv5

- **map.tas.fromPeriod=2021-01**

  Si map.tas.toV5=true: Depuis cette période de déclaration les déclarations de salaire sont "mappées" en message ELMv5

- **processing.default.hours=48**

  Si await.result=true. Calcule la valeur dans l'élément ExpectedAvailability dans la structure de réponse Processing.

- **la.splitting=false**

  Certificat de salaire: Si cette option est true, un fichier XML par collaborateur est écrit dans le zip.

- **or.enabled=false**

  Participation de collaborateur: Si cette option est true, les messages sont sauvegardés dans le système de fichiers.

- **or.splitting=false**

  Participation de collaborateur : Si cette option est true, un fichier XML par collaborateur est écrit dans le zip.

- **uc.splitting=false**

  Attestation de chômage : Si cette option est true, un fichier XML par collaborateur est écrit dans le zip.

- **write.original.xml=false**

  Si le message est fractionné et cette option est à true, aussi l'original du message de salaire est écrit dans le zip.

5.2.1.2. Configurations supplémentaires

Des configurations supplémentaires peuvent être changées dans les properties suivants.

- **housekeeping.schedule=0 */15 * * * ?**

Valeur Cron de combien de fois le Housekeeping Job est démarré. (wikipedia.org/wiki/Cron [https://de.wikipedia.org/wiki/Cron])

La valeur par défaut correspond à tous les 15 minutes

- **housekeeping.elm.enabled=true**

  Activation du Housekeeping Job

- **housekeeping.elm.days=365**

  Combien de temps les déclarations de salaire doivent-elles être sauvegardées dans la base de données. (Valeur minimale est 90 Tage)

- **backup.derby.onstartup=true**

  Est-ce que la base de données est sauvegardée quand l'application est démarrée

- **cleanup.derby.onstartup=true**

  Indexation de la base de données pour optimiser l'espace de stockage

- **monitoring.enabled=false**

  Activation de la Monitoring Console / interface REST

- **monitoring.user=admin**

  Nom d'utilisateur pour Monitoring Console

- **monitoring.pass=admin**

  Mot de passe pour Monitoring Console

5.2.1.3. Configuration du système

Ces properties servent à la configuration générale du système

- **server.port=9090**

  Le port sur lequel l'application SwissdecAdapter-Integration démarre

- **derby.system.home=C:/swissdecAdapter/derbydb**

  La base de données Derby se trouve dans ce répertoire.

- **spring.datasource.url=jdbc:derby:C:/swissdecAdapter/derbydb/step**

  Lien auprès de la base de données Derby

- **released.filesystem.parent.xml=C:/swissdecAdapter/data/received**

  Les déclarations de salaire traitées sont sauvegardées dans ce répertoire.

  (voir Tabelle 2.1, „Chemins de l'interface Swissdec-Adapter - sM-Client“)

- **failed.filesystem.parent.xml=C:/swissdecAdapter/data/failed**

  Les déclarations de salaire qui ont échoué sont sauvegardées dans ce classeur.

  (voir Tabelle 2.1, „Chemins de l'interface Swissdec-Adapter - sM-Client“)

- **result.filesystem.parent.xml=C:/swissdecAdapter/data/result**

  Les fichiers de réponse sont attendus dans ce répertoire (si await.result=true)

  (voir Tabelle 2.1, „Chemins de l'interface Swissdec-Adapter - sM-Client“)

- **undeliverable.filesystem.parent.xml=C:/swissdecAdapter/data/undeliverable**

Ces données de réponse n'ont pas pu être récupérées

(voir Tabelle 2.1, „Chemins de l'interface Swissdec-Adapter - sM-Client“)

- **maintenance.filesystem.parent.xml=C:/swissdecAdapter/data/maintenance**

  Ce dossier contient le XML de la fenêtre de maintenance.

- **logging.level.ch.itserve.step=INFO**

  Réglage du niveau LOG (Possibilitées : DEBUG, INFO, WARN, ERROR)

- **logging.file=log/swissdecAdapter-integration.log**

  Fichier LOG standard (rotation quotidienne)

- **logging.file.max.history=60**

  Après combien de jours les vieux fichiers log sont supprimés

5.2.2. Configuration de référence

```
                #
# Copyright 1996-2024 itServe AG. All rights reserved.
#
# This software is the proprietary information of itServe AG
# Bern Switzerland. Use is subject to license terms.
#


#
# Copyright 1996-2023 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

# TaxAtSource Settings
await.result=false
commune.splitting=false
testcase.auto.quittance=false
map.tas.toV5=false
map.tas.fromPeriod=2021-01
processing.default.hours=48

# Tax Settings
la.splitting=false

# Ownership Right Details
or.enabled=true
or.splitting=true

# Unemployment Certificate Settings
uc.splitting=true

# Global Settings
write.original.xml=false

# Directory Settings
released.filesystem.parent.xml=C:/swissdecAdapter/data/received
failed.filesystem.parent.xml=C:/swissdecAdapter/data/failed
result.filesystem.parent.xml=C:/swissdecAdapter/data/result
sent.filesystem.parent.xml=C:/swissdecAdapter/data/sent
undeliverable.filesystem.parent.xml=C:/swissdecAdapter/data/undeliverable
maintenance.filesystem.parent.xml=C:/swissdecAdapter/data/maintenance

# Database Settings
derby.system.home=C:/swissdecAdapter/derbydb
```

```
spring.datasource.url=jdbc:derby:C:/swissdecAdapter/derbydb/step

# Housekeeping Settings
housekeeping.schedule=0 */15 * * * ?
housekeeping.elm.enabled=true
housekeeping.elm.days=365

# Startup Settings
backup.derby.onstartup=true
cleanup.derby.onstartup=true

# Monitoring Settings
monitoring.enabled=false
monitoring.user=admin
monitoring.pass=admin

# LOG Settings
logging.level.ch.itserve.step=INFO
logging.file.name=log/swissdecAdapter-integration.log
logging.file.max-history=60
```

Exemple 5.2. integration.properties

6. Sécurité

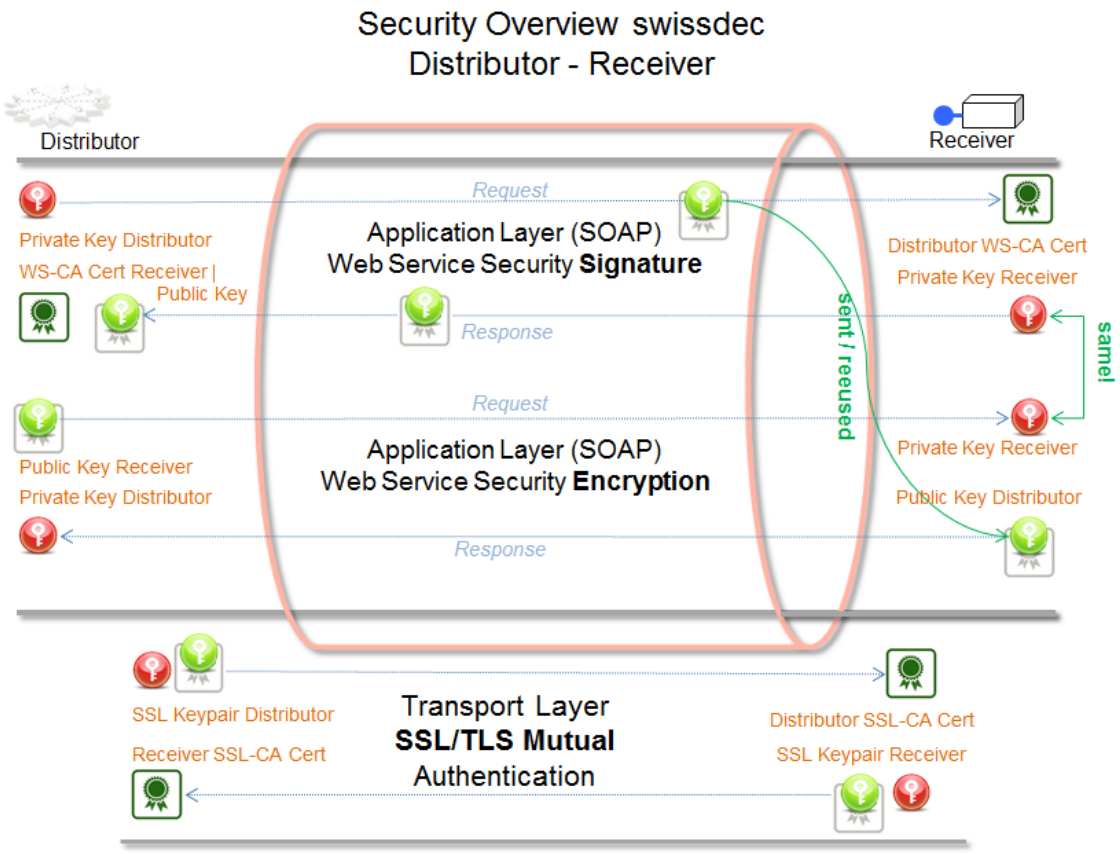


Abbildung 6.1. Aperçu de la sécurité

	Clef privée (Private Key)
	Certificat avec clef publique (Public Key). Dans ce qui suit, "clef publique" (public key) est synonymes au certificat qui contient la clef publique.
	Certificat CA

Tabelle 6.1. Légende de la figure Security Overview

6.1. Transport Layer TLS

Le client établit une connexion avec le serveur. En général, le serveur s'authentifie d'abord auprès du client à l'aide d'un certificat. Ensuite, soit le client envoie au serveur un code aléatoire avec la clé publique du serveur, soit les deux parties calculent la clé de cryptage à l'aide de l'algorithme échange de clés Diffie-Hellman un secret commun. À partir de ce secret, une clé cryptographique est dérivée. Cette clé est ensuite utilisée pour crypter tous les messages de la connexion à l'aide d'une méthode symétrique pour protéger l'intégrité des messages et l'authenticité par un code d'authentification de message.

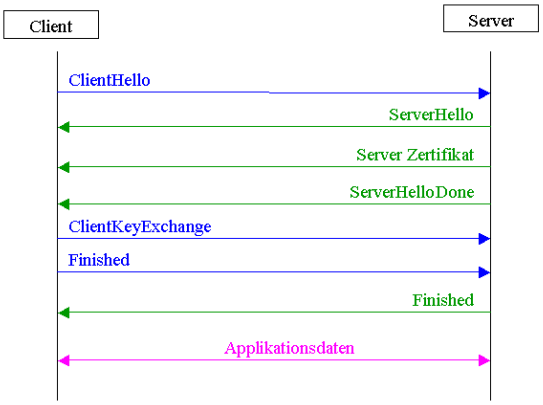


Abbildung 6.2. TLS Handshake

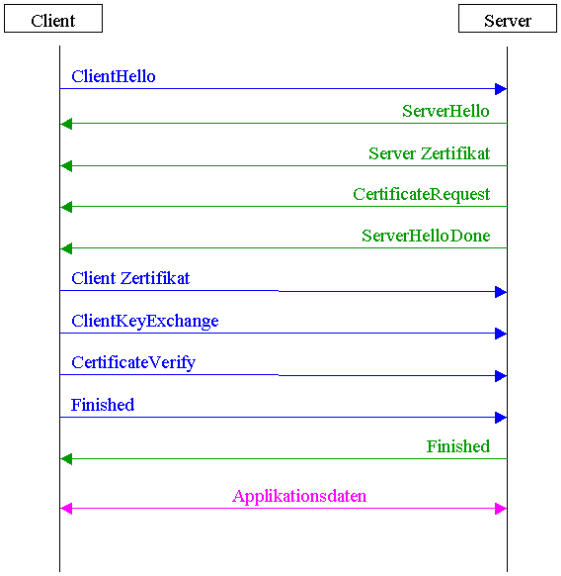


Abbildung 6.3. SSL Handshake avec Mutual Authentication

Dans le cas de l'authentification mutuelle (Mutual Authentication), le serveur exige du client que celui-ci s'authentifie également avec un certificat. Cette procédure est prescrite par la norme Swissdec afin de garantir que toutes les parties se fassent mutuellement confiance.

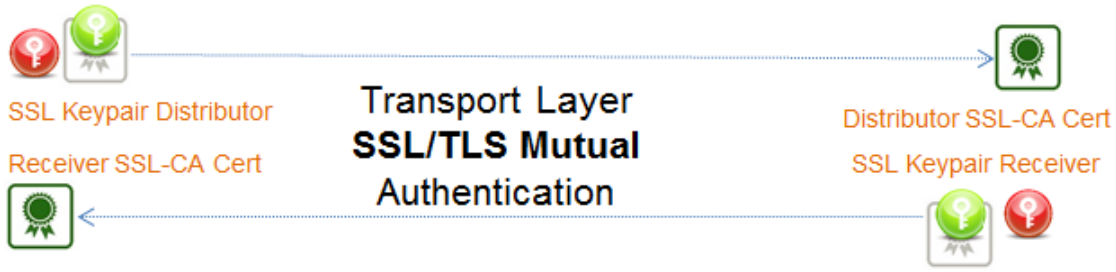


Abbildung 6.4. Transport Layer Security Overview

Dans la vue simplifiée, on voit que des deux côtés (client/Swissdec distributeur, serveur/adaptateur Swissdec), une paire de clés (clé privée et clé publique correspondante) est nécessaire pour s'identifier, ainsi que le certificat CA de l'autre partie est nécessaire pour la vérification.

La configuration TLS (???) pour le Swissdec-Adapter se trouve dans le fichier "receiver.properties". Si la terminaison doit être effectuée sur l'adaptateur un exemple de configuration se trouve dans ce fichier.

Remarques

Par simplicité *Keystore* (avec la paire de clefs) et Truststore (avec le certificat CA du distributeur Swissdec) sont fourni dans un fichier. Ainsi, les deux indications sont identiques !

Les certificats productifs peuvent être commandé par le support sM-Client [https://smcsupport.atos-solutions.ch]. Pour celà le hostname (Enregistrement DNS), avec lequel l'installation du Swissdec-Adapter est atteignable depuis l'internet, doit être déclaré.

6.2. Webservice Security

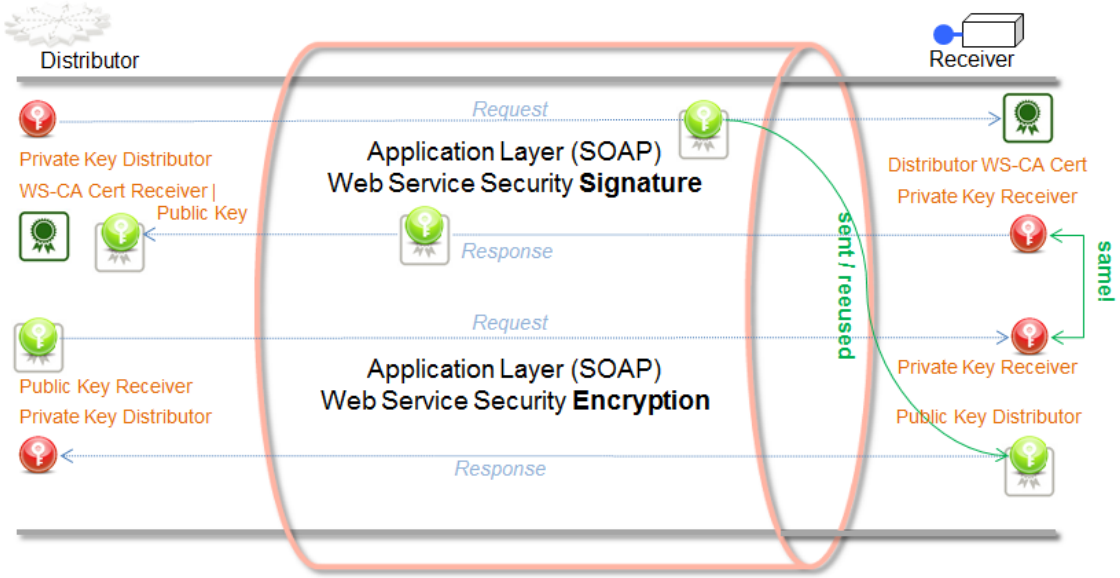


Abbildung 6.5. Webservice Security Overview

6.2.1. Signature

Request	Le client (Swissdec Distributor) signe la requête avec sa clé privée (Private Key) et envoie sa clé publique (Public Key) avec la requête. Du côté du serveur (adaptateur Swissdec) la clé publique envoyée est comparée à la clé Swissdec CA et la signature est vérifiée. Côté serveur, que le certificat Swissdec CA est nécessaire pour vérifier la signature.
Response	L'adaptateur Swissdec signe la réponse avec sa clé privée. Le distributeur Swissdec doit connaitre la clé publique de celui, respectivement le certificat CA.

Tabelle 6.2. Webservice Security Signature

Pour le processus Webservice Security Signatur de côté serveur, le certificat CA du client ansi la propre paire clefs (clef privée et publique) est nécessaire.

6.2.2. Cryptage

Request	Le client (Swissdec Distributor) crypte la requête avec la clé publique (Key Public Key) du destinataire. Du côté du serveur (adaptateur Swissdec), la requête est décryptée avec sa propre clé privée.
Response	L'adaptateur Swissdec crypte la réponse à l'aide de la clé publique du distributeur Swissdec, qui est incluse dans le request pour la signature de Webservice Security. Cela permet de garantir que seul le détenteur de la clé privée correspondante peut décrypter la réponse.

Tabelle 6.3. Webservice Security Encryption

Pour le processus de chiffrement de sécurité du service web, seule la clé privée du destinataire est requise côté serveur, qui est la même que celle utilisée pour la signature du service web.

6.2.3. Configuration

Il ressort de l'illustration 3.2 que, côté serveur (adaptateur Swissdec), deux choses sont nécessaires pour la sécurité des services web:

- Swissdec CA-certificat
- Paire de clefs (clefs privées et publiques) du destinataire

Comme le certificat CA doit seulement être connu, il n'est pas configuré explicitement. Il suffit qu'il soit présent dans le *keystore* configuré. La clé privée du destinataire final doit toutefois être explicitement adressée dans le *keystore*. Les options de configuration suivantes (???) (application.properties) contrôlent le comportement de l'adaptateur Swissdec pour la sécurité des services web.

Remarque

Les certificats productifs peuvent être commandé par le support sM-Client [https://smcsupport.atos-solutions.ch].

Dès que les certificats de sécurité des services web pour la production sont installés, aucune transmission ne peut plus être acceptée de la part de la Refapps et du TestTool. Ceci parce que les signatures des messages ne peuvent plus être vérifiées, car le distributeur de production utilise des certificats d'une autre CA (Certificate Authority).

7. Fonctionnement

7.1. Configuration du logging

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

Option	Default	Description
logging.level.ch.itserve.step	INFO	Log Level de tous les messages du paquet ch.itserve.step
logging.file	log/swissdecAdapter-receiver.log	Le fichier dans lequel les logs sont écrit.
logging.file.max-history	1	Indique combien de jours les fichiers log sont sauvegardés.

Tabelle 7.1. Configuration Log

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

7.2. Fenêtre de maintenance

Comme les messages sont transmis de manière synchrone (en temps réel) sur la plate-forme Swissdec, les clients voient, lors de la transmission, quels sont les destinataires finaux qui n'étaient pas joignables. Il est possible d'annoncer des interruptions de service planifiées au moyen de fenêtres de maintenance. Dans la mesure du possible, celles-ci devraient être planifiées entre 20h00 et 6h00 en semaine ou le week-end. Cela permet de fournir aux clients une notification plutôt qu'une erreur. Pour enregistrer une fenêtre de maintenance, procédez comme suit:

1. Créez un XML "PlannedMaintenance". Dans le répertoire integration/templates vous trouverez un modèle (PlannedMaintenance.xml).

2. Enregistrez-le dans l'application d'intégration de l'adaptateur Swissdec dans le répertoire "maintenance.filesystem.parent.xml". Le nom n'a pas d'importance, l'extension du fichier doit être ".xml".

3. La fenêtre de maintenance sera communiquée lors du prochain ping du distributeur. L'accessibilité du destinataire final est contrôlée depuis le distributeur de manière cyclique (toutes les 30 minutes). Pour ce faire, l'opération de service web "PingConsumer" du destinataire final est appelée, qui confirme à son tour l'accessibilité en répondant. Dans ce processus, il est possible d'inclure dans la réponse du destinataire final des fenêtres de maintenance. Ces fenêtres sont enregistrées sur le distributeur.

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

Remarque

- Le nome du fichier n'a aucune importance. La terminaison doit être impérativement ".xml".
- Les heures "Start" et "End" dans le "PlannedMaintenance" XML sont du type xs:dateTime. Cela signifie que le fuseau horaire et l'heure d'été/d'hiver doivent également y figure.
  - 2013-08-31T13:40:47.0Z: le Z signifie UTC
  - 2014-01-31T13:40:47.0+01:00: UTC +1 heure, correspond à heure de l'Europe centrale (heure d'hiver)
  - 2013-08-31T13:40:47.0+02:00: UTC +2 heures, correspond à l'heure d'été de l'Europe centrale
- Les "messages" sont communiqués jusqu'à la comptabilité salariale. Veuillez choisir des textes raisonnables dans les 3 langues (allemand, français, italien).
- Il est possible de placer plusieurs XML "PlannedMaintenance" dans le répertoire "maintenance.filesystem.parent.xml" . L'adaptateur Swissdec va toujours annoncer le suivant au distributeur et, à l'expiration, le supprimer automatiquement resp. signaler le suivant.

Exemple d'un "PlannedMaintenance" XML:

```
        <?xml version="1.0" encoding="UTF-8"?>
<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>
```

7.3. Monitoring

Le Swissdec-Adapter peut être surveillé comme tout autre processus ou service standard. Si le Swissdec-Adapter a été enregistré comme service sous Windows, la „Windows Computer Management Console“ peut être utilisée pour vérifier/modifier l'état du service.

7.3.1. Monitoring intégré

Le monitoring intégré au moyen de l'interface REST est expliquée dans le chapitre Console d'administration.

7.4. Console d'administration

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

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

L'interface RESTful est atteignable à l'URL suivante:

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

Lors de la consultation, veuillez vous assurer que l'en-tête de requête HTTP "Accept : application/json" est défini. L'authentification est assurée au moyen de HTTP-BASIC. Exemple d'un request:

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

Concernant l'authentification, vous trouvez plus d'information sous Wikipedia [https://en.wikipedia.org/wiki/Basic_access_authentication]

7.4.1. Resources

7.4.1.1. Statistiques

Aperçu:

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

7.4.1.2. Monitoring

Atteignable à l'URL:

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

Exemple de sortie du monitoring intégré :


```
{

   "connectivity": {
      "lastpingDateTime": "20.12.2016 09:37:44",
      "lastpingTimestamp": "1482223064619"
   },
   "database": {
      "path": "derbydb",
      "version": "3.0_4 (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": {
      ...
   }

}
```

A. Annexe

A.1. Apache Reverse Proxy

Ces instructions d'installation sont données à titre indicatif et peuvent être adaptées en fonction des spécifications du système.

Dans cet exemple, le point final de swissdec.be.ch est configuré avec le certificat C:/Certificats/server.crt et la clé C:/Certificats/server.key . Le SwissdecAdapter Receiver fonctionne sur le même système sur le port 8080.

Le certificat mutual se trouve dans C:/Zertifikate/mutual.pem

1. Download Apache d'ApacheHaus (apachehaus.com) (Apache 2.4.x OpenSSL 1.1.1 VC15)

2. Extraire dans des dossiers (Exemple : C:/Apache24)

3. Installer Apache comme service et démarrer:

```
C:/Apache24/bin/httpd -k install
C:/Apache24/bin/httpd -k start
```

4. Configuration de C:/Apache24/conf/httpd.conf

   charger mod_proxy et mod_proxy_http (enlever le commentaire "#")

```
LoadModule proxy_module modules/mod_proxy.so
LoadModule proxy_http_module modules/mod_proxy_http.so
```

5. Configuration de SSL dans C:/Apache24/conf/extra/httpd-ahssl.conf

```
<VirtualHost swissdec.be.ch:443>
   SSLEngine on
   ServerName swissdec.be.ch:443
   SSLCertificateFile "C:/Zertifikate/server.crt"
   SSLCertificateKeyFile "C:/Zertifikate/server.key"
   # Les deux lignes suivantes sont utilisées pour Mutual Authentication
   SSLCACertificateFile "C:/Zertifikate/mutual.pem"
   SSLVerifyClient require
   DocumentRoot "${SRVROOT}/htdocs"
   # DocumentRoot access handled globally in httpd.conf
   CustomLog "${SRVROOT}/logs/ssl_request.log" "%t %h %{SSL_PROTOCOL}x %{SSL_CIPHER}x \"%r\" %b"
      <Directory "${SRVROOT}/htdocs">
         Options Indexes Includes FollowSymLinks
         AllowOverride AuthConfig Limit FileInfo
         Require all granted
      </Directory>
   ProxyPass         /    http://localhost:8080/
   ProxyPassReverse  /    http://localhost:8080/
</VirtualHost>
```

6. Redémarrez Apache Service

Si les certificats sont seulement en format JKS, ils peuvent être exportés relativement facilement dans le format désiré avec le Keystore Explorer (https://keystore-explorer.org/) .

A.2. Testtool

Pour pouvoir vérifier l'installation, Swissdec met à disposition un TestTool. Il s'agit d'un simple client de service web qui permet d'envoyer des décomptes d'impôt à la source et des certificats de salaire au format Swissdec ELM v4.0 et ELM v5.x.

A.2.1. Installation

Le TestTool est livré sous forme de fichier ZIP qu'il suffit de décompresser. Pour l'exécuter, une installation de Java est nécessaire.

A.2.2. Configuration

Avant de tester, les fichiers

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

doivent être adaptées. Il s'agit des fichiers properties au format XML.

Option	Description
endpointURL	L'adresse à tester du Swissdec-Adapters.
wssEncryptionCert	Le certificat est utilisé pour le cryptage des messages (Public Key des SwissdecAdapters).
sigrootcert	Le certificat est utilisé pour vérifier la signature de la réponse (Public Key des SwissdecAdapters).
sslcert	Certificat avec le Public Key pour le TLS-Private Key
sslkey	TLS-Private Key pour l'authentification du client
sslpartnercacert	CA-certificat pour vérifier le certificat SSL/TLS du serveur.

Tabelle A.1. Configuration du TestTool

Plus d'information concernant la sécurité est disponible dans le document Kapitel 6, *Sécurité*.

Warnung

La configuration fournie correspond à la configuration default du Swissdec-Adapter.

L'option "sslpartnercacert" est utilisé seulement si HTTPS est activé.

Les options "sslcert" et "sslkey" sont utilisées seulement si Mutual Authentication est activé.

A.2.3. Execution des tests

Pour exécuter un test, le script batch ou le script shell correspondant est exécuté. Une documentation détaillée sur les cas de test est fournie avec TestTool.

A.3. Tableau unicode

Les caractères spéciaux dans institution.name doivent être remplacés

Exemple pour "Steuerverwaltung Kt. Graubünden" est échangé avec:

```
institution.name=Steuerverwaltung Kt. Graub\u00FCnden
```

Caractère spécial	Unicode
Ä	\u00C4
ä	\u00E4
Ö	\u00D6
ö	\u00F6
Ü	\u00DC
ü	\u00FC
é	\u00E9
è	\u00E8
ç	\u00E7

Tabelle A.2. Tableau unicode pour les caractères spéciaux

A.4. Documents référencés

[ENDRECREQ] *Directives pour la transmission des données salariales*. Exigences relatives au destinataire final. Swissdec. Version 5 (20200220).

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

A.5. Glossaire

DMZ	un réseau informatique avec des possibilités d'accès contrôlées en termes de sécurité aux serveurs qui y sont connectés
Keystore	Un keystore est une base de données de clés et de certificats dans un fichier.
NTP	Norme pour la synchronisation des horloges dans les systèmes informatiques.
sM-Client	Le sM-Client offre des fonctionnalités permettant d'envoyer et de recevoir des messages dans différents formats.
STEPrx	Produit de l'entreprise itServe AG, receveur *Swissdec* pour recevoir des messages Swissdec.
Swissdec	Label de qualité pour les systèmes de comptabilité salariale. Standards XML pour la norme en matière de salaire-CH (ELM, Einheitliche Lohn-Meldung).