

Sips Office Server 2.04 **Guide d'installation**

Version 2.01 – Octobre 2010



REACH YOUR TARGETS >>

Contact

By phone: +33 (0)811 107 033
By fax: +33 (0)811 107 033
By email: sips@atosorigin.com

Avertissements :

Le fichier version.txt de l'API Sips Office Server précise l'environnement dans lequel il a été compilé et testé. Il précise également les versions minimales des composants à utiliser avec cette version de l'API Sips Office Server.

L'installation de l'API sur tout autre environnement n'est pas garantie.

Sommaire

1. INTRODUCTION.....	3
2. ARCHITECTURE DE L'API SIPS OFFICE SERVER	4
3. LISTE DES OBJETS LIVRES	5
4. INSTALLATION DU KIT API SIPS OFFICE SERVER	7
4.1 INSTALLATION DES OBJETS.....	7
4.2 CONFIGURATION DES PARAMETRES.....	7
4.2.1 Fichier config.xml par défaut.....	8
4.2.2 Configuration des paramètres de service	9
4.2.3 Configuration des paramètres de commande	10
4.2.4 Configuration des paramètres de l'accueil http d'administration	11
4.2.5 Configuration des paramètres du watchdog.....	13
4.2.6 Configuration de l'URL d'alerte :	15
4.2.7 Configuration des paramètres des composants	16
4.3 CONFIGURATION DES LISTES D'ACCES.....	17
4.4 FICHIERS QUI NE DOIVENT PAS ETRE MODIFIES	19
5. LANCEMENT DU SERVEUR	20
5.1 SCRIPTS DE LANCEMENT	20
5.2 EDITION DES SCRIPTS.....	20
5.2.1 OS Windows.....	20
5.2.2 OS Unix.....	20
5.3 LANCEMENT DE L'API SIPS OFFICE SERVER.....	21
5.4 ERREURS AU LANCEMENT DE L'API SIPS OFFICE SERVER	21
6. DETAIL DE L'AJOUT ET DE LA MISE A JOUR D'UN COMPOSANT	22
6.1 MISE A JOUR	22
6.2 AJOUT D'UN COMPOSANT	22
7. DESINSTALLATION	23
7.1 DESINSTALLATION D'UN COMPOSANT	23
7.2 DESINSTALLATION DE L'API SIPS OFFICE SERVER.....	23

1. INTRODUCTION

Ce document vous explique comment installer l'API Sips Office Server sur votre serveur. L'installation devant être effectuée sur la machine qui doit héberger l'API Sips Office Server, il faut bien entendu posséder des droits d'administration. En effet, il faut pouvoir installer (si non présent) un JRE (Java Runtime Environment nécessaire à l'exécution de programmes Java, disponible en libre téléchargement sur internet), lancer et/ou arrêter l'API Sips Office Server, et éditer manuellement certains fichiers.

Note : Une connexion FTP est insuffisante, il faut au minimum une liaison telnet sur le serveur final.

Ce document vous montre également de façon générale comment installer un ou plusieurs composants sur API Sips Office Server. Le paramétrage spécifique des composants n'est pas décrit ici, reportez vous au *GUIDE DU COMPOSANT* adapté.

Note : Ce document ne décrit pas comment interfacier votre système existant avec l'API Sips Office Server. L'interface (XML) des différents services offerts par ce logiciel est décrite dans le *GUIDE DU DEVELOPPEUR*.

Convention d'écriture

Dans tout le document, les conventions d'écriture suivantes seront utilisées :

- Les renvois à d'autres documentations seront notés en majuscules et en italique.

ex : *GUIDE DU DEVELOPPEUR*

- Les différents champs des fichiers de configuration seront notés en italique :

ex : *<service></service>*

Pré-requis pour l'installation :

Outre les droits d'administration décrits ci-avant, des connaissances réseau sont indispensables pour exécuter correctement l'installation.

2. ARCHITECTURE DE L'API SIPS OFFICE SERVER

L'API Sips Office Server est basé sur une architecture JAVA. Il consiste en deux processus distincts qui se surveillent mutuellement. Ces deux processus partagent les mêmes archives (fichiers .jar) et les mêmes fichiers de configuration, mais ils ont chacun leurs fonctionnalités.

Le Serveur d'accueil constitue le premier processus. Ce serveur a les fonctionnalités suivantes :

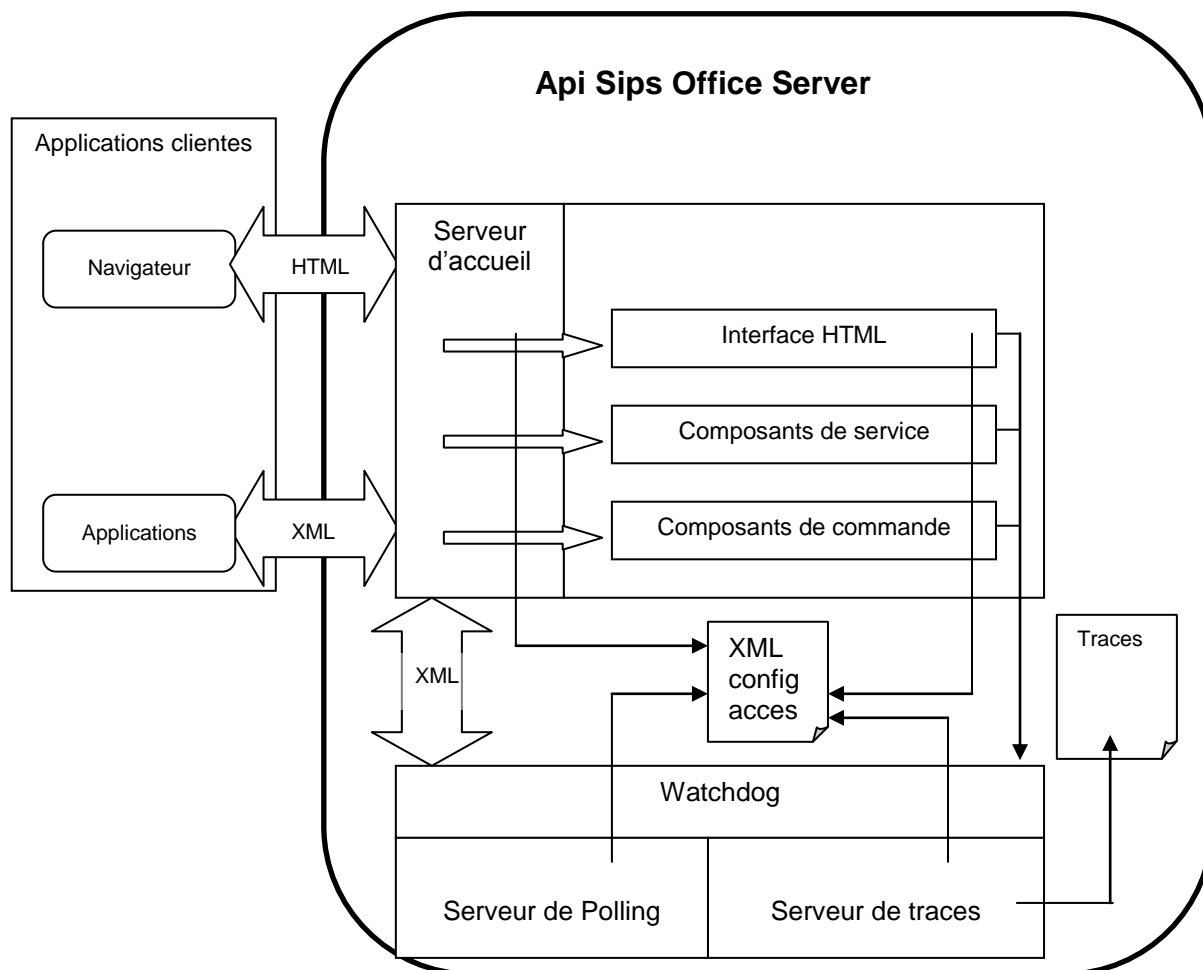
- hébergement d'une interface HTML d'administration
- Gestion des accès client au serveur
- hébergement de composants de services et de commande
- Accueil, contrôle et routage des requêtes client vers les composants

Le second processus est appelé watchdog. Il gère les fonctionnalités suivantes :

- Serveur RMI (Remote Method Invocation) de traces pour les autres processus.
- Serveur de Polling qui surveille le serveur d'accueil

Les étapes d'installation décrites dans ce document vous permettront d'installer complètement le bloc API Sips Office Server du schéma ci-après :

Schéma d'architecture :



3. LISTE DES OBJETS LIVRES

Les fichiers suivants sont livrés dans l'archive SIPS_APISERVER_204.tar.

Répertoire server/access	fichiers de configuration des droits d'accès
accessList.dtd	fichier dtd contenant la description des accessList
accessListCommand.xml	Liste d'accès aux requêtes de commandes
accessListHttpRequest.xml	Liste d'accès à l'interface d'administration
accessListService.xml	Liste d'accès aux requêtes de services
password.conf	Fichier contenant le mot de passe d'administration crypté
Répertoire server/components/command	Archives des composants de commande
password.jar	archive du composant password
register.jar	archive du composant register
status.jar	archive du composant status
traces.jar	archive du composant de traces
Répertoire server/components/service	Archives des composants de service
Ce répertoire est vide, il sera complété par l'installation des composants	
Répertoire server/config	fichiers de configuration principaux
config.dtd	descriptif du fichier de configuration xml
config.xml	fichier de configuration principal de l'application API Sips Office Server
Répertoire server/HttpRoot	Racine de l'interface d'administration HTML
Répertoire Html	<u>Ces fichiers ne doivent pas être modifiés.</u>
Répertoire Xsl	
Répertoire server/jaxp1.1	Archives standard Java nécessaires
crimson.jar	
jaxp.jar	
xalan.jar	
Répertoire server/lib	Archives principales de l'application API Sips Office Server
apiServer.jar	
Répertoire server/rmi	fichier de configuration RMI
client.policy	fichier policy pour les requêtes rmi, <u>à ne pas modifier</u>

Répertoire server/script	scripts de lancement de l'API Sips Office Server
launch.bat	fichier de lancement de l'API Sips Office Server (WINDOWS)
launch.sh	fichier de lancement de l'API Sips Office Server (Unix)
startAPIServer.bat	fichier de lancement du serveur d'accueil seul (WINDOWS)
startAPIServer.sh	fichier de lancement du serveur d'accueil seul (Unix)
startWatchdog.bat	fichier de lancement du watchdog seul (WINDOWS)
startWatchdog.sh	fichier de lancement du watchdog seul (Unix)
Répertoire server/traces	Répertoire des fichiers de traces
Répertoire exemples/	Exemples de requêtes getStatus.
Asp	Exemple Asp
C_Solaris_Linux	Exemple C Solaris et Linux
C_Windows	Exemple C Windows
Java	Exemple Java
Perl	Exemple Perl
Php	Exemple Php

4. INSTALLATION DU KIT API SIPS OFFICE SERVER

Cette première phase a pour but d'installer l'API Sips Office Server sur votre machine. On ne décrit pas ici l'installation de composants de services, mais uniquement l'installation du serveur autonome qui permettra d'héberger ces composants.

L'application API Sips Office Server est écrite totalement en Java. Vous pouvez par conséquent l'installer sur n'importe quel OS supportant un JRE de version 1.3 ou supérieure.

L'installation du JRE n'est pas présentée ici.

La commande « `java -version` » vous donnera la version de JRE java installée sur votre machine dans le cas où vous en possédez déjà une. Si aucun JRE n'est installé, ou si la version est antérieure à 1.3, consultez le site de sun <http://java.sun.com/> pour télécharger et installer un JRE de version suffisante.

4.1 INSTALLATION DES OBJETS

Tout d'abord vous devez installer les fichiers de l'API Sips Office Server :

Pour cela, copiez le fichier SIPS_APISERVER_204.tar livré sur le disque dur de votre serveur.

Ensuite, pour un OS de type Unix : détarrer le fichier SIPS_APISERVER_204.tar en utilisant la commande suivante :

```
tar -xvf SIPS_APISERVER_204.tar
```

Sous Windows : décompressez SIPS_APISERVER_204.tar dans le répertoire où vous souhaitez installer API Sips Office Server à l'aide d'un logiciel de compression qui accepte les fichiers dont l'extension est « .tar ».

Vous devez alors retrouver l'arborescence présentée au chapitre précédent.

4.2 CONFIGURATION DES PARAMETRES

Tous les paramètres de l'API Sips Office Server sont situés dans le fichier config.xml. Un certain nombre de ces paramètres sont positionnés par défaut et sont donc modifiables à votre convenance.

Note : Une fois l'API Sips Office Server installé, toute modification des paramètres de configuration nécessitera un arrêt/relance de l'API Sips Office Server pour qu'elle soit prise en compte.

La configuration par défaut n'est pas suffisante pour faire fonctionner l'API Sips Office Server, il faudra donc nécessairement éditer le fichier config.xml, pour modifier au moins les paramètres suivants :

- `<accessListService>`
- `<accessListCommand>`
- `<accessListHttpRequest>`
- `<documentRoot>`
- l'attribut *path* des éléments `<trace>` et `<alternateTrace>`
- `<componentDir>`

4.2.1 Fichier config.xml par défaut

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE configuration SYSTEM "config.dtd">
<configuration>
  <service>
    <servicePort>7180</servicePort>
    <maxServiceRequest>5</maxServiceRequest>
    <queueSizeMaxService>5</queueSizeMaxService>
    <accessListService>chemin_du_fichier_accessListService.xml</accessListService>
  </service>
  <command>
    <commandPort>7181</commandPort>
    <maxCommandRequest>5</maxCommandRequest>
    <queueSizeMaxCommand>5</queueSizeMaxCommand>
    <accessListCommand>chemin_du_fichier_accessListCommand.xml
    </accessListCommand>
  </command>
  <httpRequest>
    <httpRequestPort>7182</httpRequestPort>
    <maxHttpRequest>1</maxHttpRequest>
    <queueSizeMaxHttpRequest>1</queueSizeMaxHttpRequest>
    <accessListHttpRequest>chemin_du_fichier_accessListHttpRequest.xml
    </accessListHttpRequest>
  </httpRequest>
  <documentRoot>chemin_du_repertoire_HttpRoot</documentRoot>
  <watchdog>
    <pollingTimer>6000</pollingTimer>
    <survPort>7183</survPort>
    <trace level="0" sizeLimit="1000" unit="Line" path="chemin_du_repertoire_trace"
prefix="APIServer" />
  <!--
    <alternateTrace id="example" path="chemin_du_repertoire_trace" prefix="example" />
  -->
    <alternateTrace id="" path="." prefix=""
  </watchdog>
  <routing>
    <componentDir>chemin_du_repertoire_components</componentDir>
  </routing>
  <alert>
    <alertURL>http://</alertURL>
    <alertProxyHost></alertProxyHost>
    <alertProxyPort></alertProxyPort>
  </alert>
  <components>
    <pathfile id="" path="chemin_du_fichier_pathfile" />
  </components>
</configuration>
```

4.2.2 Configuration des paramètres de service

Les paramètres de service concernent tous les composants service que vous installerez avec l'application API Sips Office Server. Ces paramètres sont :

- le port de service.
- le nombre maximum de requêtes traitables en parallèle.
- le nombre maximum de requêtes à stocker dans les queues des messages.
- le chemin du fichier de configuration des droits d'accès.

Ces données déterminent la gestion de la file d'attente des services.

Vous trouverez ci-dessous le paragraphe concerné dans le fichier config.xml :

```
<service>
  <servicePort>7180</servicePort>
  <maxServiceRequest>5</maxServiceRequest>
  <queueSizeMaxService>5</queueSizeMaxService>
  <accessListService>chemin_du_fichier_accessListService.xml
</accessListService>
</service>
```

<servicePort> : c'est le numéro de port sur lequel les requêtes de service vont être dirigées. C'est aussi par ce port que vont être retournées les réponses (connexion socket).

<maxServiceRequest> : le serveur que vous allez installer permet de fixer un nombre maximum de requêtes de service à traiter en parallèle. Ceci dans le but de ne pas trop surcharger la machine. Les requêtes reçues une fois ce nombre atteint sont stockées dans une file d'attente.

<queueSizeMaxService> : définit la taille des files d'attente (Il y a une file d'attente pour chacun des processus parallèles définis ci-avant). Les requêtes reçues une fois ce nombre atteint ne sont pas prises en compte et sont perdues. En cas de dépassement de capacité, une alerte indique cet état.

Note : Une requête placée en file d'attente occupe environ 10 Ko en mémoire.

Ces trois valeurs doivent être des entiers. Des valeurs par défaut vous sont proposées. Le numéro de port a été choisi de façon à ne pas créer de conflit avec d'autres applications courantes installées sur la machine. Les autres données dépendent de la configuration matérielle utilisée.

<accessListService> : ce dernier champ doit être paramétré en fonction du répertoire où vous avez installé l'API Sips Office Server. La description du fichier accessListService.xml est faite au paragraphe 4.3

Avec un OS Windows, par exemple :

```
<accessListService>C:\server\access\accessListService.xml</accessListService>
```

Avec un OS de type Unix, par exemple :

```
<accessListService>/home/server/access/accessListService.xml</accessListService>
```

4.2.3 Configuration des paramètres de commande

Les paramètres de commande concernent tous les composants commande installés par défaut avec l'application API Sips Office Server. Ces paramètres sont :

- le port de commande.
- le nombre maximum de requêtes traitables en parallèle.
- le nombre maximum de requêtes à stocker dans les queues des messages.
- le chemin du fichier de configuration des droits d'accès.

Ces données déterminent la gestion de la file d'attente des commandes.

Vous trouverez ci-dessous le paragraphe concerné dans le fichier config.xml :

```
<command>
  <commandPort>7181</commandPort>
  <maxCommandRequest>5</maxCommandRequest>
  <queueSizeMaxCommand>5</queueSizeMaxCommand>
  <accessListCommand>chemin_du_fichier_accessListCommand.xml
</accessListCommand>
</command>
```

<commandPort> : c'est le numéro de port sur lequel les requêtes de commande vont être dirigées. C'est aussi par ce port que vont être retournées les réponses (connexion socket).

<maxCommandRequest> : le serveur que vous allez installer permet de fixer un nombre maximum de requêtes de commande à traiter en parallèle. Ceci dans le but de ne pas trop surcharger la machine. Les requêtes reçues une fois ce nombre atteint sont stockées dans une file d'attente.

<queueSizeMaxCommand> : définit la taille de la file d'attente (Il y a une file d'attente pour chacun des processus parallèles définis ci-avant). Les requêtes reçues une fois ce nombre atteint ne sont pas prises en compte et sont perdues. En cas de dépassement de capacité, une alerte indique cet état.

Note : Une requête placée en file d'attente occupe environ 10 Ko en mémoire.

Ces trois valeurs doivent être des entiers. Des valeurs par défaut vous sont proposées. Le numéro de port a été choisi de façon à ne pas créer de conflit avec d'autres applications courantes installées sur la machine. Les autres données dépendent de la configuration matérielle utilisée.

<accessListCommand> : ce dernier champ doit être paramétré en fonction du répertoire où vous avez installé l'API Sips Office Server. La description du fichier accessListCommand.xml est faite au paragraphe 4.3.

Avec un OS Windows, par exemple :

```
<accessListCommand>C:\chemin\server\access\accessListCommand.xml</accessListCommand>
```

Avec un OS de type Unix, par exemple :

```
<accessListCommand>/home/server/access/accessListCommand.xml</accessListCommand>
```

4.2.4 Configuration des paramètres de l'accueil http d'administration

Les paramètres de l'accueil Http concernent l'interface Html d'administration de l'application API Sips Office Server. Ces paramètres sont :

- le port d'écoute du serveur Http.
- le nombre maximum de requêtes traitables en parallèle.
- le nombre maximum de requêtes à stocker dans les queues des messages.
- le chemin du fichier de configuration des droits d'accès.

Ces données déterminent la gestion et la réception des requêtes de l'interface Html d'administration :

Vous trouverez ci-dessous le paragraphe concerné dans le fichier config.xml :

```
<httpRequest>
  <httpRequestPort>7182</httpRequestPort>
  <maxHttpRequest>1</maxHttpRequest>
  <queueSizeMaxHttpRequest>1</queueSizeMaxHttpRequest>
  <accessListHttpRequest>chemin_du_fichier_accessListHttpRequest.xml
</accessListHttpRequest>
</httpRequest>
```

<httpRequestPort> : c'est le numéro de port sur lequel vous devrez vous connecter à l'aide d'un navigateur afin d'ouvrir la page d'administration. Pour plus de détails sur la console HTML, référez vous au *GUIDE D'ADMINISTRATION*.

<maxHttpRequest> : le serveur permet de fixer un nombre maximum de requêtes Http à traiter en parallèle. Les requêtes reçues une fois ce nombre atteint sont stockées dans une file d'attente.

<queueSizeMaxHttpRequest> : définit la taille de cette file d'attente (Il y a une file d'attente pour chacun des processus parallèles définis ci-avant). Les requêtes reçues une fois ce nombre atteint ne sont pas prises en compte et sont perdues.

Note : Une requête placée en file d'attente occupe environ 10 Ko en mémoire. Notez que le paramétrage de ce chapitre concerne l'accès à l'interface HTML d'administration, qui ne doit en principe être utilisée que par l'administrateur, d'où les valeurs par défaut à 1.

Ces trois valeurs doivent être des entiers. Le numéro de port par défaut a été choisi de façon à ne pas créer de conflit avec d'autres applications courantes installées sur la machine. Il est fortement déconseillé d'utiliser le port standard HTTP (80) qui est probablement déjà en service pour votre serveur web. Les autres données dépendent de la configuration matérielle utilisée.

<accessListHttpRequest> : ce dernier champ doit être paramétré en fonction du répertoire où vous avez installé l'API Sips Office Server. La description du fichier accessListHttpRequest.xml est faite au paragraphe 4.3.

Avec un OS Windows, par exemple :

```
<accessListHttpRequest>C:\chemin\server\access\accessListHttpRequest.xml</accessListHttpRequest>
```

Avec un OS de type Unix, par exemple :

```
<accessListHttpRequest>/home/server/access/accessListHttpRequest.xml</accessListHttpRequest>
```

Enfin, l'interface d'administration étant ni plus ni moins qu'un serveur Http, elle présente évidemment un repertoire Http racine, également paramétré dans le fichier config.xml.

```
<documentRoot>chemin_du_repertoire_HttpRoot</documentRoot>
```

<documentRoot> doit donc être paramétré avec le chemin complet du repertoire HttpRoot, qui lui aussi dépend du repertoire dans lequel vous avez installé l'API Sips Office Server.

Avec un OS Windows, par exemple :

```
<documentRoot>C:\server\HttpRoot</documentRoot>
```

Avec un OS de type Unix, par exemple :

```
<documentRoot>/home/server/HttpRoot</documentRoot>
```

4.2.5 Configuration des paramètres du watchdog

Le watchdog est un second processus serveur qui tourne en parallèle du serveur d'accueil. Ce serveur a deux fonctions principales :

- La vérification du bon fonctionnement du serveur
- La gestion des traces

Ici encore, un certain nombre de paramètres sont positionnés par défaut. Les deux champs que vous devez obligatoirement modifier sont les champs « path » des objets « trace » et « alternateTrace », qui dépendent explicitement du chemin où vous avez installé votre API Sips Office Server.

Vous trouverez ci-dessous le paragraphe concerné dans le fichier config.xml :

```
<watchdog>
  <pollingTimer>6000</pollingTimer>
  <survPort>7183</survPort>
  <trace level="0" sizeLimit="1000" unit="Line" path="chemin_du_repertoire_trace"
prefix="APIServer" />
  <alternateTrace id="" path="." prefix=""
<!-- <alternateTrace id="example" path="chemin_du_repertoire_trace" prefix="example" /> -->
</watchdog>
```

<pollingTimer> : durée entre deux tests d'activité du serveur : C'est une valeur entière exprimée en secondes. A chaque intervalle de temps correspondant à cette valeur, le watchdog va envoyer une requête de commande et tester la cohérence de la réponse

Note : Une valeur faible (y compris la valeur 0) du pollingTimer force le pollingServer à envoyer des requêtes continuellement. Ce comportement n'est pas recommandé pour un mode de production.

<survPort> : numéro du port de communication entre le serveur et le watchdog : c'est le numéro de port (RMI) sur lequel les requêtes d'écriture de traces vont être envoyées et par lequel les réponses de lecture de celles-ci vont être retournées.

Le numéro de port par défaut (7183) a été choisi de façon à ne pas créer de conflit avec d'autres applications courantes installées sur la machine

L'élément **<trace>** permet de déterminer les paramètres qui seront appliqués à toutes les traces de l'API Sips Office Server.

- Le paramètre *level* initialisé par défaut à 0 détermine le niveau de trace de l'API Sips Office Server (0 : normal, 1 : fatal, 2 : erreur, 3 : warning, 4 : debug). Selon son besoin, l'administrateur peut changer ce niveau pour des niveaux supérieurs. Notez qu'en niveau 3 et 4, les traces deviennent rapidement très volumineuses, et dégradent fortement les temps de réponse de l'API Sips Office Server. Ces niveaux ne seront donc en aucun cas adaptés à un fonctionnement en production.
- les traces générées par le serveur sont rotatives. Lorsque le fichier atteint la valeur paramétrée, celui-ci est renommé et un nouveau est créé. La valeur maximale de la taille d'un fichier de trace se paramètre grâce aux deux champs *sizeLimit* et *unit*. *unit* est l'unité de mesure qui sera utilisée pour mesurer la taille des fichiers de traces. Elle peut prendre deux valeurs, soit « Line » pour ligne, ou « Ko » pour KiloOctet. *sizeLimit* est une valeur entière qui donne le nombre maximal de valeurs unitaires qui peut être contenu dans un fichier de traces. Ainsi les valeurs par défaut produiront des fichiers de traces d'une taille maximale de 1000 lignes.

- Le champ *path* doit contenir le chemin du répertoire des traces de votre application API Sips Office Server. Ce dernier dépend du répertoire dans lequel vous avez installé votre application.

Avec un OS Windows, par exemple :

```
path="C:\chemin_racine\server\trace"
```

Avec un OS de type Unix, par exemple :

```
path="/home/chemin_racine/server/trace"
```

- Le champ *prefix* vous permet de paramétrer le préfixe du fichier de traces standard d'utilisation de l'API Sips Office Server.

```
<!-- <alternateTrace id="example" path="chemin_du_repertoire_trace" prefix="example" /> -->
```

<alternateTrace> sert à configurer les traces spécifiques des différents composants que vous installerez avec API Sips Office Server. La ligne ci-dessus est donnée à titre d'exemple et doit être laissée en commentaire.

```
<alternateTrace id="" path="." prefix=""
```

La seconde ligne présentée ci-dessus est indispensable pour permettre le démarrage de l'API Sips Office Server sans installer de composant additionnel.

- On doit trouver dans le fichier config.xml une ligne *alternateTrace* par composant de service installé. La balise *alternateTrace* est rattachée au composant grâce au champ « *id* », qui doit contenir l'identifiant du composant, soit le nom de l'archive de service sans l'extension « .jar ».

Les traces des composants suivront nécessairement les mêmes règles de rotate et de niveau que les traces standard. Pour le paramétrage des traces des composants, référez-vous au *GUIDE DU COMPOSANT* correspondant.

4.2.6 Configuration de l'URL d'alerte :

Comme expliqué auparavant, l'application API Sips Office Server comprend deux processus qui se surveillent mutuellement. Si un problème est détecté lors de cette surveillance, un message est envoyé en méthode POST sur l'URL d'alerte paramétrée dans le fichier config.xml. Ces messages et leurs causes sont décrits dans le *GUIDE D'ADMINISTRATION*.

L'utilisation de cette fonctionnalité n'est pas obligatoire. Le paramétrage explicité ci-après n'est donc pas indispensable pour faire fonctionner l'API Sips Office Server, mais il faut malgré tout laisser en place la configuration par défaut de l'élément <alert>.

Vous trouverez ci-dessous le paragraphe concerné dans le fichier config.xml :

```
<alert>
  <alertURL></alertURL>
  <alertProxyHost></alertProxyHost>
  <alertProxyPort></alertProxyPort>
</alert>
```

<alertURL> : cet élément vous permet de paramétrer l'url du cgi, de la servlet, ou autre fonction chargée du traitement des alertes.

Par exemple

```
<alertURL>http://monServeur.com/cgi-bin/alert.cgi</alertURL>
```

Si vous ne souhaitez pas utiliser cette fonctionnalité, il vous suffit de laisser la configuration en place, c'est-à-dire sans URL.

<alertProxyHost> et <alertProxyPort> : si vous devez passer par un proxy entre l'API Sips Office Server et la machine qui héberge votre script de traitement des alertes, il vous suffit de paramétrer ces deux champs supplémentaires.

Par exemple, pour passer par le proxy monProxy de port 8080.

```
<alertProxyHost>monProxy</alertProxyHost>
```

```
<alertProxyPort>8080</alertProxyPort>
```

Si vous n'utilisez pas de proxy, laissez en place la configuration par défaut de ces deux champs.

Note : Il n'est pas possible de paramétrer un user/password pour passer par ce proxy. Le cas échéant, il faudra agir sur la configuration du proxy lui-même.

L'utilisation de l'URL d'alerte est détaillée dans le *GUIDE D'ADMINISTRATION*.

4.2.7 Configuration des paramètres des composants

Ces paramètres, présentés ci-dessous, servent au fonctionnement des composants de façon générale.

```
<routing>
  <componentDir>chemin_du_repertoire_composants</componentDir>
</routing>
<components>
  <pathfile id="" path="" />
</components>
```

Vous devrez dans tous les cas paramétrer la balise `<routing>` avec le chemin du répertoire des composants de votre serveur. Attention, le chemin à renseigner ici est le chemin du répertoire parent des répertoires des composants de commande et de services. Dans les objets livrés, ce répertoire est nommé composants.

Avec un OS Windows, par exemple :

```
<componentDir>C:\chemin\server\composants</componentDir>
```

Avec un OS de type Unix, par exemple :

```
<componentDir>/home/server/composants</componentDir>
```

Par contre, selon les composants que vous installerez, vous devrez peut-être modifier certains paramètres de la balise composants. Les détails de ces modifications ne sont pas explicités ici, car ils dépendent directement du composant que vous installez. Reportez-vous au *GUIDE DU COMPOSANT* que vous souhaitez installer pour connaître les paramètres à renseigner.

4.3 CONFIGURATION DES LISTES D'ACCES

Attention, ces listes d'accès sont très importantes, ce sont elles qui vont définir quelles machines vont avoir le droit d'accéder aux différentes parties de l'API Sips Office Server. En effet il en existe pour l'accès à l'accueil http de l'interface d'administration, aux services et aux commandes. Elles sont toutes basées sur le même schéma. Vous devrez nécessairement paramétrer ces fichiers pour pouvoir accéder aux fonctionnalités de l'API Sips Office Server depuis toute autre machine que celle hébergeant l'application elle-même.

Les noms de ces fichiers sont explicites :

- accessListService.xml détermine les droits d'accès aux requêtes de service.
- accessListCommand.xml détermine les droits d'accès aux requêtes de commande.
- accessListHttpRequest détermine les droits d'accès à l'interface Html d'administration.

Ces fichiers sont stockés dans le répertoire « server/access » du répertoire principal de l'API Sips Office Server.

Ces fichiers ont tous le même format XML, présenté ci-dessous :

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE list SYSTEM "accessList.dtd">
<list>
  <accept>
    <acceptList from="1.1.1.1" to="1.1.1.20" />
    <acceptList from="1.1.2.1" to="1.1.2.11" />
    <acceptAddress>8.8.8.8</acceptAddress>
    <acceptAddress>5.5.5.5</acceptAddress>
  </accept>
  <deny>
    <denyList from="1.1.1.3" to="1.1.1.10" />
    <denyList from="2.2.2.3" to="2.2.2.11" />
    <denyAddress>8.8.8.8</denyAddress>
    <denyAddress>9.9.9.9</denyAddress>
  </deny>
</list>
```

Note : Les éléments <list></list>, <accept></accept> et <deny></deny> sont obligatoires. Ils doivent être présents dans le fichier d'accès même si ils ne contiennent aucun élément.

D'autre part, les éléments <acceptList>, <acceptAddress>, <denyList> et <denyAddress> ne doivent pas être vides. Pour supprimer une règle, il faut supprimer la ligne complète.

Un manquement à l'une de ces règles empêche le démarrage de l'API Sips Office Server au moment du chargement de l'accessList concernée.

Les règles de gestion des accès sont les suivantes :

- Tout ce qui n'est pas autorisé est interdit.
- Le deny a priorité sur l'accept (cf. l'exemple précédent, 8.8.8.8 n'est pas accepté).
- Le localhost (127.0.0.1) est autorisé par défaut même s'il ne figure pas dans la liste.

La syntaxe des différents champs est la suivante :

<acceptList> : Si vous souhaitez donner l'accès à une plage d'adresse particulière cela se fera au travers de cet élément, qui prend en attribut « from » et « to ».

Par exemple pour un accès de la machine 104.3.2.1 à la machine 104.3.2.20, vous devez insérer la ligne `<acceptList from="104.3.2.1" to="104.3.2.20" />` entre les lignes `<accept>` et `</accept>`. Les règles d'établissement des plages sont semblables à celles des Firewalls, mais la notion de masque n'existe pas. Ainsi, la ligne `<acceptList from="104.3.2.0" to="104.3.4.0" />` laissera passer 2 x 256 + 1 adresses (les adresses 104.3.2.*, 104.3.3.*, et l'adresse 104.3.4.0).

<acceptAddress> : Si vous souhaitez maintenant ajouter l'accès à une machine particulière, vous allez utiliser cet élément.

Par exemple l'ajout de la machine d'adresse 104.3.2.44 se fera en insérant la ligne `<acceptAddress>104.3.2.44</acceptAddress>` entre les balises `<accept>` et `</accept>`.

<denyList> : Sur le même principe que l'ajout, vous pouvez interdire une plage d'adresse au travers de cet élément, qui comme pour l'ajout d'une liste prend en attribut « from » et « to ». Par exemple pour interdire l'accès aux machine d'adresse 104.3.2.3 à 104.3.2.10, vous devez insérer la ligne `<denyList from="104.3.2.3" to="104.3.2.10" />` entre les lignes `<deny>` et `</deny>`.

<denyAddress> : Enfin, sur le même principe que l'ajout d'accès à une machine, vous pouvez interdire des accès au travers de cet élément. Par exemple l'interdiction de la machine d'adresse 104.3.2.74 se fera en insérant la ligne `<denyAddress>104.3.2.74</denyAddress>` entre les balises `<deny>` et `</deny>`.

Note : Les listes d'accès peuvent être modifiées selon les besoins des différentes applications clientes. Les règles de gestion restent bien entendu les mêmes que pour l'installation. Comme l'objet de contrôle d'accès est instancié et initialisé au démarrage de l'API Sips Office Server, un arrêt/relance de l'API Sips Office Server est indispensable pour prendre en compte ces modifications.

4.4 FICHIERS QUI NE DOIVENT PAS ETRE MODIFIES

- Dans le répertoire « access » du dossier « server » est présent un fichier nommé « password.conf ». Celui-ci contient le résultat du chiffrement du login et du mot de passe d'accès à l'accueil Http. Il doit toujours être présent à cet emplacement dans ce répertoire et ne doit pas être modifié manuellement. Le login/password chiffré par défaut est admin/admin. Vous aurez la possibilité de le modifier par la suite grâce à l'interface d'administration HTML de l'application API Sips Office Server, ou en mode batch.
- Dans les différents répertoires présents sur votre machine à la suite d'une installation, vous pourrez remarquer des fichiers dont l'extension est « .dtd ». Ces fichiers ne contiennent pas d'informations sur la configuration de votre application mais ils servent à contrôler la structure des documents XML. Ces fichiers ne doivent en aucun cas être modifiés. Tout changement entraînera un dysfonctionnement logique de l'application.
- Les archives java correspondant aux services et aux commandes doivent rester dans leurs dossiers respectifs au sein du répertoire « components ».

5. LANCEMENT DU SERVEUR

5.1 SCRIPTS DE LANCEMENT

Dans le répertoire script du répertoire principal de l'application API Sips Office Server se trouvent différents scripts de lancement.

Les fichiers .bat sont dédiés aux environnements de type Windows

Les fichiers .sh sont dédiés aux environnements de type Unix, Linux, etc

Les scripts launch permettent de lancer les deux processus Serveur de composants et Watchdog avec la même commande.

Les scripts startServer et startWatchdog permettent de ne lancer qu'un seul des deux serveurs à la fois.

Ces fichiers ne sont pas complètement configurés par défaut, il faut les éditer pour prendre en compte l'arborescence de votre serveur.

5.2 EDITION DES SCRIPTS

5.2.1 OS Windows

Les scripts de lancement à éditer sont les fichiers dont l'extension est .bat.

Editez le fichier launch.bat.

La valeur par défaut « SERVER_ROOT=c:\server\ » suppose que vous ayez installé API Sips Office Server à la racine du lecteur C de votre machine. Cette variable doit contenir le chemin complet du répertoire server de l'application.

Modifiez la valeur de SERVER_ROOT en fonction du répertoire où vous avez installé votre API Sips Office Server.

Enregistrez et fermez le fichier launch.bat

Vous devez faire de même avec tous les autres scripts de lancement que vous souhaitez utiliser.

5.2.2 OS Unix

Les scripts de lancement à éditer sont les fichiers dont l'extension est .sh.

Editez le fichier launch.sh.

La valeur par défaut SERVER_ROOT= « /home/server/ » suppose que vous ayez installé API Sips Office Server directement sous le répertoire /home/ de votre machine. Cette variable doit contenir le chemin complet du répertoire server de l'application.

Modifiez la valeur de SERVER_ROOT en fonction du répertoire où vous avez installé votre API Sips Office Server.

Enregistrez et fermez le fichier launch.sh

Vous devez faire de même avec tous les autres scripts de lancement que vous souhaitez utiliser.

5.3 LANCEMENT DE L'API SIPS OFFICE SERVER

Vérifiez dans un premier temps que les droits d'exécution des scripts de lancement que vous avez édités sont corrects. Vous pouvez ensuite lancer le script launch soit en ligne de commande, soit directement depuis l'interface graphique de votre OS.

L'apparition dans votre console du message « Serveur en attente d'invocations clientes » vous signale que le serveur d'accueil a correctement démarré. Il faut encore attendre une vingtaine de secondes avant que tous les autres processus soient initialisés.

Votre API Sips Office Server est maintenant complètement installé. Vous pouvez le vérifier en vous connectant à l'interface HTML d'administration comme expliqué dans le *GUIDE D'ADMINISTRATION*. Le user/mot de passe installé par défaut est admin/admin.

5.4 ERREURS AU LANCEMENT DE L'API SIPS OFFICE SERVER

Des erreurs peuvent se présenter lors du démarrage :

Exemple de messages	Cause	solution
'java' n'est pas reconnu en tant que commande interne ou externe, un programme exécutable ou un fichier de commandes. (WNT) ou « not found » (UNIX)	Le binaire java n'est pas accessible	renseignez le chemin complet du binaire java dans vos scripts de démarrage
java.lang.ClassNotFoundException: at java.lang.Class.forName(Class.java:52)	La version de JRE utilisée est insuffisante.	Assurez-vous d'utiliser un JRE de version 1.3 ou supérieure.
java.lang.ClassNotFoundException: "nom de classe non trouvée"	La version de JRE utilisée est insuffisante.	Assurez-vous d'utiliser un JRE de version 1.3 ou supérieure.
java.lang.ClassNotFoundException: at com.atosorigin.services....	Le script ne peut accéder à l'archive apiServer.jar	Vérifier les droits de lecture de l'archive, et qu'elle n'est pas corrompue. Vérifier le chemin de la variable SERVER_ROOT de votre script de démarrage.
java.lang.ClassNotFoundException: at org.w3c.....	Le script ne peut accéder à l'archive de la classe désignée	Vérifier les droits de lecture de l'archive (répertoire jaxp1.1 ou jre), et qu'elle n'est pas corrompue.
Server Dead or echo Port not responding	Une erreur empêche le serveur de démarrer.	Observez les traces et corrigez l'erreur le cas échéant.
Autres messages		Contactez le Centre d'Assistance Technique

6. DETAIL DE L'AJOUT ET DE LA MISE A JOUR D'UN COMPOSANT

6.1 MISE A JOUR

La procédure de mise à jour d'un composant est particulièrement simple. Il suffit d'arrêter le l'API Sips Office Server comme expliqué dans le *GUIDE D'ADMINISTRATION*, de remplacer dans le répertoire composants/service/ le jar installé par le nouveau jar livré, et de redémarrer le serveur. Si la mise à jour nécessite du paramétrage particulier, celui-ci vous sera signalé avec le Kit de mise à jour.

6.2 AJOUT D'UN COMPOSANT

La procédure d'installation d'un composant est détaillée dans le *GUIDE DU COMPOSANT* associé.

De façon générale, le composant sera constitué d'un fichier jar à placer sous « composants/service », et de fichiers de paramètres placés dans un répertoire « composant » dans le répertoire d'installation d'API Sips Office Server, au même niveau que le répertoire server.

Il faudra ensuite paramétrer le fichier config.xml pour qu'il reconnaisse ce nouveau composant, en ajoutant une ligne de type alternateTrace.

```
<alternateTrace id="composant" path="chemin" prefix="fichier" />
```

avec

composant : le nom du nouveau composant installé

chemin : le chemin du répertoire de traces

fichier : le nom du prefixe des fichiers de traces pour ce composant

Il faudra peut-être ajouter des éléments entre les lignes <component> et </component>, selon le type du composant.

Toutes ces modifications sont présentées en détail dans le *GUIDE DU COMPOSANT*.

Enfin, il faudra effectuer l'enregistrement du nouveau composant, soit par l'interface d'administration, soit directement en lançant une requête batch d'enregistrement comme expliqué dans le *GUIDE D'ADMINISTRATION*.

Note : Dans le cas d'un ajout de nouveau composant, il n'est pas nécessaire de faire un arrêt/relance de l'API Sips Office Server.

7. DESINSTALLATION

7.1 DESINSTALLATION D'UN COMPOSANT

Référez vous au *GUIDE DU COMPOSANT* associé.

De façon générale, il faudra arrêter l'API Sips Office Server, supprimer le jar du composant du répertoire components/service/, supprimer les lignes de paramétrages ajoutés à config.xml lors de l'installation du composant, supprimer le répertoire des paramètres du composant, et enfin relancer API Sips Office Server.

7.2 DESINSTALLATION DE L'API SIPS OFFICE SERVER

La désinstallation est particulièrement simple. Il suffit d'arrêter l'API Sips Office Server comme expliqué dans le *GUIDE D'ADMINISTRATION*, puis de supprimer le répertoire racine de l'API Sips Office Server. En effet, comme vous avez pu le constater lors de l'installation, tous les fichiers de l'API Sips Office Server sont situés dans le même répertoire, à moins que vous n'ayez vous même déplacé certains d'entre eux.

Il vous appartient par la suite de supprimer ou non le JRE présent sur la machine, suivant l'utilité que vous en avez.