

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

Version 2.00 – 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

Sommaire

1. INTRODUCTION.....	3
2. COMPRENDRE L'API SIPS OFFICE SERVER	4
3. INTERFACE D'ADMINISTRATION	6
3.1 ACCES A LA CONSOLE D'ADMINISTRATION	6
3.1.1 Connexion à la console d'administration	6
3.1.2 Saisie du mot de passe	7
3.2 UTILISATION DE LA CONSOLE D'ADMINISTRATION	8
3.2.1 La commande « VERIFIER L'ETAT DU SERVEUR »	8
3.2.2 La commande « LIRE LES TRACES »	9
3.2.3 La commande « LISTER LES COMPOSANTS »	10
3.2.4 La commande « ENREGISTRER UN NOUVEAU COMPOSANT »	11
3.2.5 La commande « VOIR LA CONFIGURATION DU SERVEUR »	12
3.2.6 La commande « CHANGER LE MOT DE PASSE D'ADMINISTRATION »	13
3.2.7 La commande « ARRETER LE SERVEUR »	14
4. ADMINISTRER EN MODE BATCH	15
4.1 SECURITE.....	15
4.2 LA COMMANDE « VERIFIER L'ETAT DU SERVEUR »	16
4.2.1 Paramètres de la requête.....	16
4.2.2 Paramètres de la réponse.....	16
4.2.3 Trame de requête XML.....	17
4.2.4 Exemple de trame de réponse XML	17
4.3 LA COMMANDE « LIRE LES TRACES »	18
4.3.1 Paramètres de la requête.....	18
4.3.2 Paramètres de la réponse.....	18
4.3.3 Trame de requête XML.....	18
4.3.4 Exemple de réponse	18
4.4 LA COMMANDE « LISTER LES COMPOSANTS »	19
4.4.1 Paramètres de la requête.....	19
4.4.2 Paramètres de la réponse.....	19
4.4.3 Trame de requête XML.....	20
4.4.4 Exemple de réponse	20
4.5 LA COMMANDE « ENREGISTRER UN NOUVEAU COMPOSANT »	21
4.5.1 Paramètres de la requête.....	21
4.5.2 Paramètres de la réponse.....	21
4.5.3 Trame de requête XML.....	21
4.5.4 Exemple de réponse	21
4.6 LA COMMANDE « CHANGER LE MOT DE PASSE D'ADMINISTRATION »	22
4.6.1 Paramètres de la requête.....	22
4.6.2 Paramètres de la réponse.....	22
4.6.3 Trame de requête XML.....	22
4.6.4 Trame de réponse XML.....	22
4.7 LA COMMANDE « ARRETER LE SERVEUR »	23
4.7.1 Paramètres de la requête.....	23
4.7.2 Paramètres de la réponse.....	23
4.7.3 Trames de requête XML.....	23
4.8 ABSENCE DE REPONSE ET MESSAGES D'ERREUR.....	24
5. DESCRIPTION DES TRACES	25

5.1 FORMAT DES TRACES	25
5.2 NIVEAUX DE TRACES.....	25
5.3 TRACES DE L'API SIPS OFFICE SERVER	26
6. REMONTEE D'ALERTE	27

1. INTRODUCTION

Ce guide est destiné à l'administrateur de l'API Sips Office Server. Il montre comment administrer ce serveur, soit en utilisant l'interface HTML, soit en mode batch.

Pré-requis :

Le kit API Sips Office Server doit avoir été installé au préalable. Vous devez connaître le login/password d'administration ainsi que le répertoire d'installation du kit API Sips Office Server.

Administration batch :

Vous devez lire le *GUIDE DU DEVELOPPEUR* pour savoir comment envoyer une requête vers l'API Sips Office Server si vous souhaitez pouvoir l'administrer en mode batch.

Conventions 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 : *LE GUIDE D'INSTALLATION*

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

ex : *<service></service>*

- Les différents éléments des requêtes et des réponses XML seront notés en gras :

ex : **component**

2. COMPRENDRE L'API SIPS OFFICE SERVER

L'API Sips Office Server est implémenté sous la forme d'un serveur autonome qui reçoit les requêtes fonctionnelles ou de commande en provenance des applications clientes et les transmet à des composants spécifiques en fonction du type de requête.

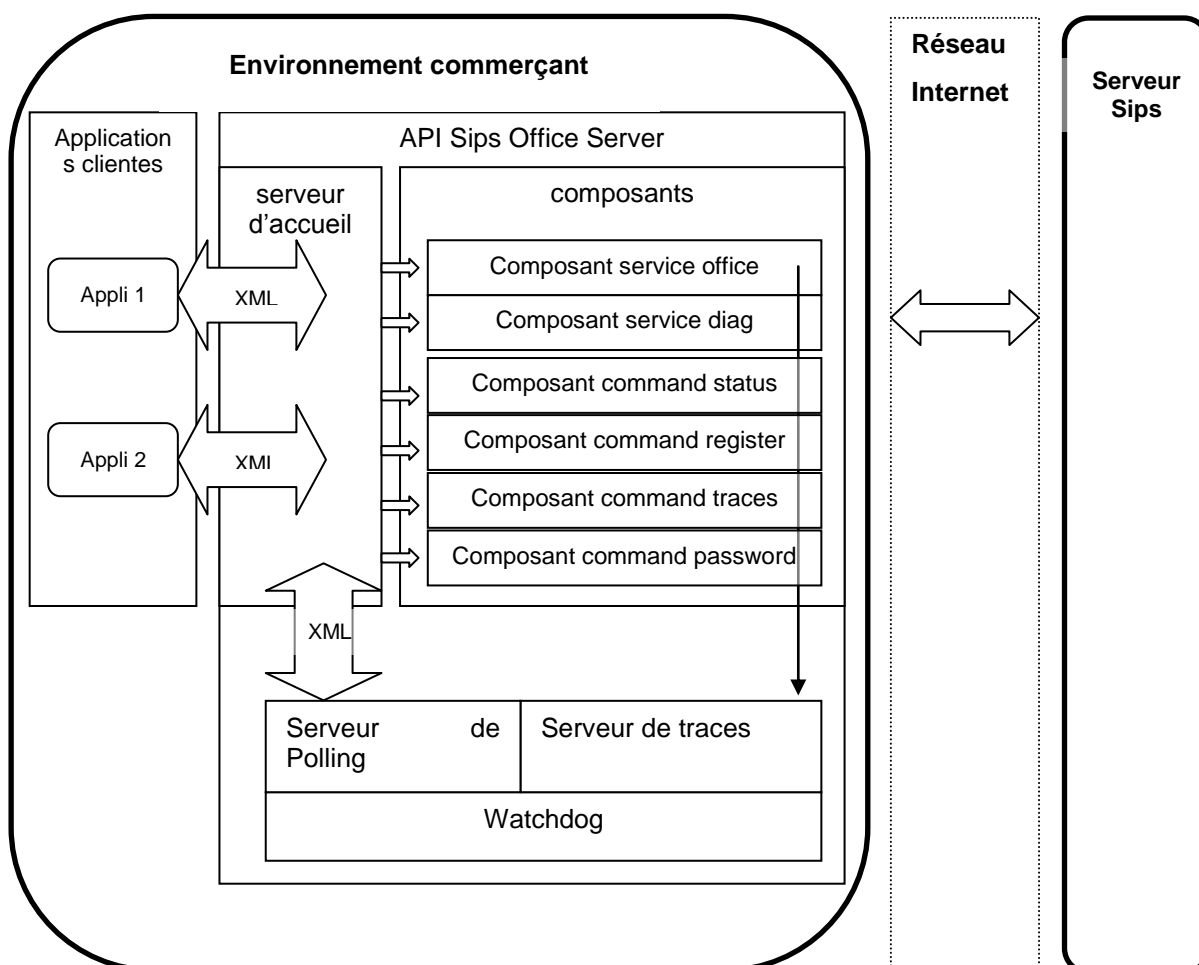
L'architecture technique de l'API Sips Office Server est décrite dans le *GUIDE DU DEVELOPPEUR*. La liste des objets livrés ainsi que leur emplacement se trouve dans le *GUIDE D'INSTALLATION*.

Surveillance de l'API Sips Office Server

L'API Sips Office Server est constitué de deux processus : le watchdog et le serveur d'accueil qui se surveillent mutuellement. Si le serveur d'accueil n'est plus disponible, comme le watchdog effectue des requêtes de polling vers l'API Sips Office Server, il enverra une requête http sur l'URL d'alerte paramétrée dans le fichier config.xml dès que l'une de ses requêtes de polling échouera.

Le watchdog est également responsable de l'écriture des traces. Si le watchdog n'est plus disponible, il n'y aura par conséquent plus de traces sur API Sips Office Server. Par contre, dès la première réception d'une requête par un composant, il y aura une tentative d'appel RMI, et une alerte http remontera sur l'URL d'alerte.

Architecture générale



Les composants

Les composants sont en charge du traitement fonctionnel correspondant aux requêtes émises par le client. Ils sont appelés par l'API Sips Office Server qui aura au préalable effectué les vérifications de syntaxe et de contenu.

Ces composants peuvent selon les cas

- Composants de service : se connecter vers les serveurs de ATOS ORIGIN pour accéder à un service distant (demande de paiement, de diagnostic) et se charger d'analyser et retourner la réponse de ces serveurs.
- Composants de commande : opérer un traitement local : c'est le cas des requêtes de commande (état du serveur, lecture et visualisation des traces ...)

Les composants de commande sont livrés en même temps que l'API Sips Office Server.

Il est possible d'ajouter, supprimer, ou remplacer un composant spécifique par la suite. La prise en compte de ce changement se réalise automatiquement sur le serveur en exécutant la commande de mise à jour des composants, ou en effectuant un arrêt/relance selon les cas.

3. INTERFACE D'ADMINISTRATION

Pour faciliter la gestion de l'API Sips Office Server, une interface graphique permet de suivre l'API Sips Office Server à distance.

3.1 ACCES A LA CONSOLE D'ADMINISTRATION

3.1.1 Connexion à la console d'administration

A partir de votre navigateur, vous devez vous connecter en tapant l'adresse suivante :

Si le serveur est installé sur votre machine :

http://localhost:httpRequestPort

Le localhost (127.0.0.1) est autorisé par défaut à se connecter à la console d'administration. Vous ne rencontrerez donc pas de problèmes de liste d'accès dans ce cas.

Sinon taper l'URL suivante (l'adresse de votre machine doit être dans la liste d'accès de l'accueil http, cf. *LE GUIDE D'INSTALLATION*) :

http://adresseIPMachineContenantServeurAPI:httpRequestPort

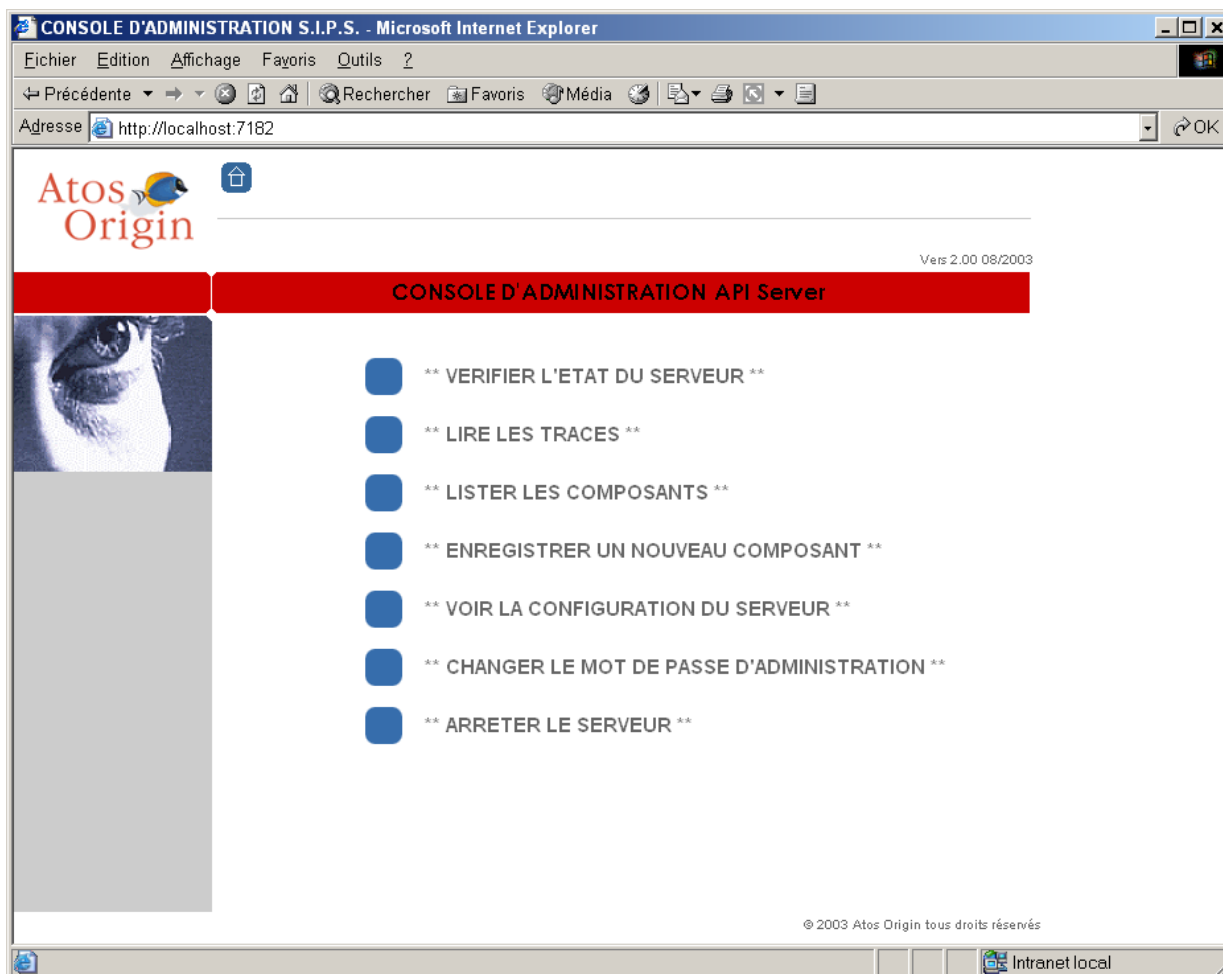
Si vous ne vous rappelez plus du numéro de port sur lequel vous devez vous connecter pour avoir accès à votre console, vous le trouverez dans le fichier config.xml derrière le tag :

`<httpRequestPort>`

Le port renseigné par défaut est le port 7182, mais il a pu être modifié lors de l'installation.

3.1.2 Saisie du mot de passe

L'accueil HTTP vous demandera de vous authentifier avant de pouvoir accéder à votre console d'administration. Vous devez donc vous munir de votre login et mot de passe (ceux paramétrés lors de l'installation de l'API Sips Office Server). Ces données sont encodées lors de l'installation de l'API Sips Office Server, et ne sont pas récupérables par la suite. La personne qui a installé l'API Sips Office Server les connaît nécessairement. Après avoir saisi le login et password, la page d'accueil suivante s'affiche :



3.2 UTILISATION DE LA CONSOLE D'ADMINISTRATION

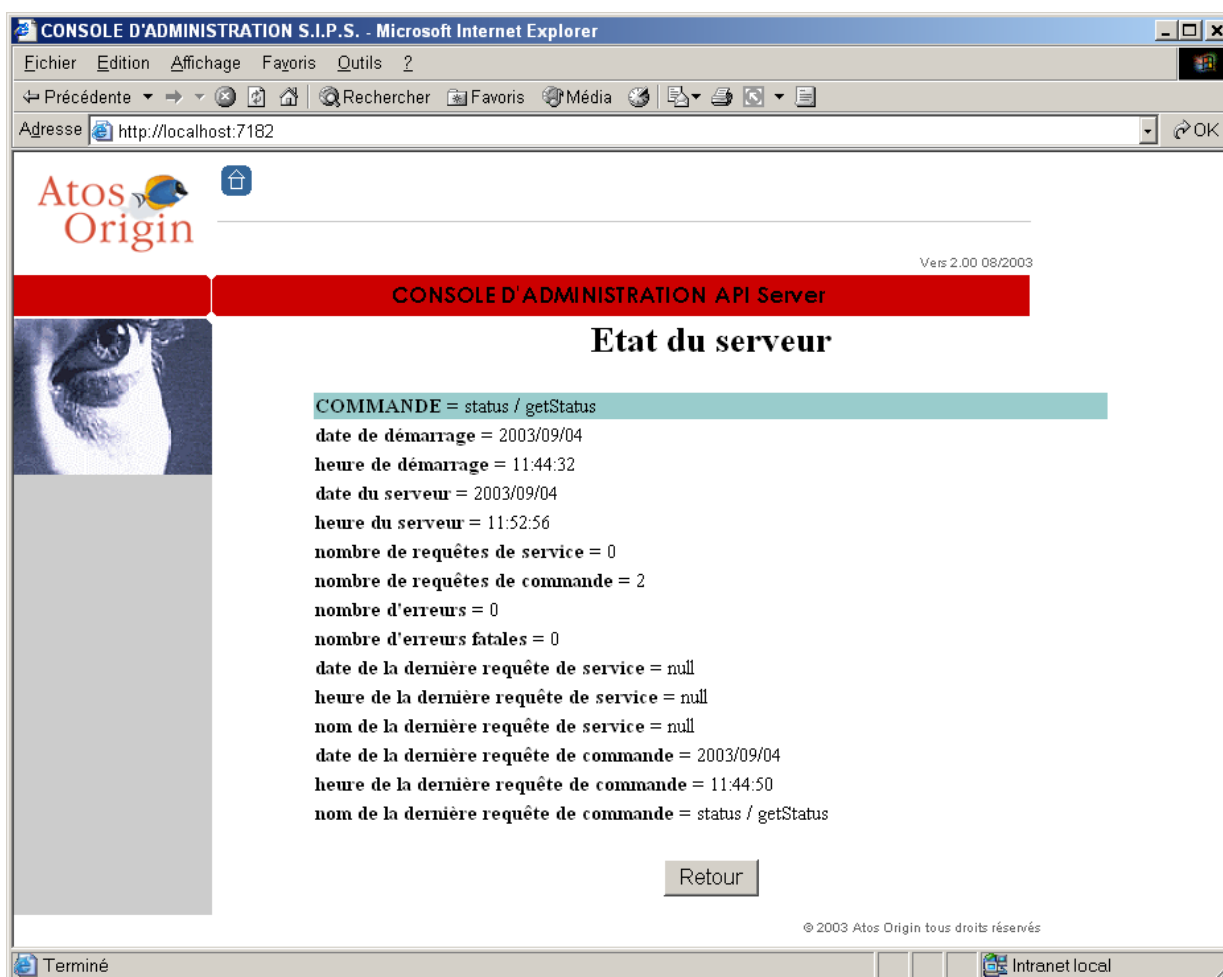
3.2.1 La commande « VERIFIER L'ETAT DU SERVEUR »

Cette commande vous permet de récupérer diverses informations sur l'état de l'API Sips Office Server, soit dans l'ordre :

- la date et l'heure de lancement du serveur
- la date et l'heure actuelle du serveur
- le nombre de demandes de service et de commande reçues depuis le lancement
- le nombre d'erreurs et d'erreurs fatales depuis le lancement
- la date, l'heure et le nom de la dernière requête de service
- la date, l'heure et le nom de la dernière demande de commande

Les traces d'erreurs et leurs différents niveaux sont décrits au chapitre 5.

Voici la page correspondante :



Note

Le nombre de requêtes de commande affiché inclut les requêtes de polling.

3.2.2 La commande « LIRE LES TRACES »

Cette commande vous permet de faire une recherche multicritère dans les fichiers de traces sur l'ensemble du serveur et des composants.

CONSOLE D'ADMINISTRATION S.I.P.S. - Microsoft Internet Explorer

Adresse <http://localhost:7182>

Atos Origin

Vers 2.00 08/2003

CONSOLE D'ADMINISTRATION API Server

Choisissez le type de traces

Date de début (AAAAMMJJ)

Date de fin (AAAAMMJJ)

Heure de début (HHMMSS)

Heure de fin (HHMMSS)

© 2003 Atos Origin tous droits réservés

Terminé Intranet local

L'« Heure de début » et l'« Heure de fin » permettent de réduire la lecture des traces sur un intervalle de chaque journée de la période définie par « Date de début » et « Date de fin ». Si les heures ne sont pas remplies, la recherche se fera sur l'ensemble de la journée. Par contre, la saisie de chacune des deux dates est obligatoire.

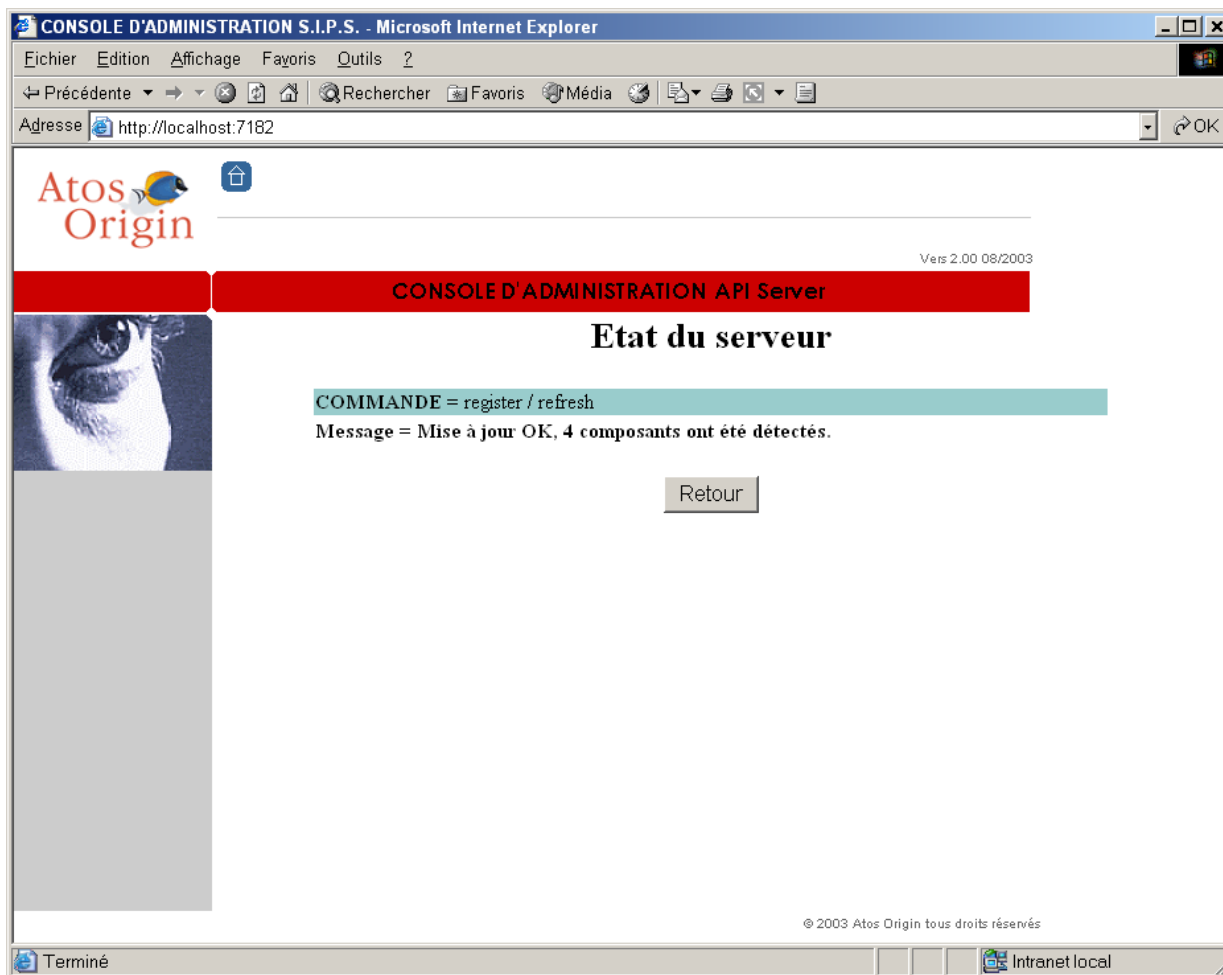
3.2.3 La commande « LISTER LES COMPOSANTS »

Vous pouvez consulter les différents composants installés sur la machine ou le serveur.



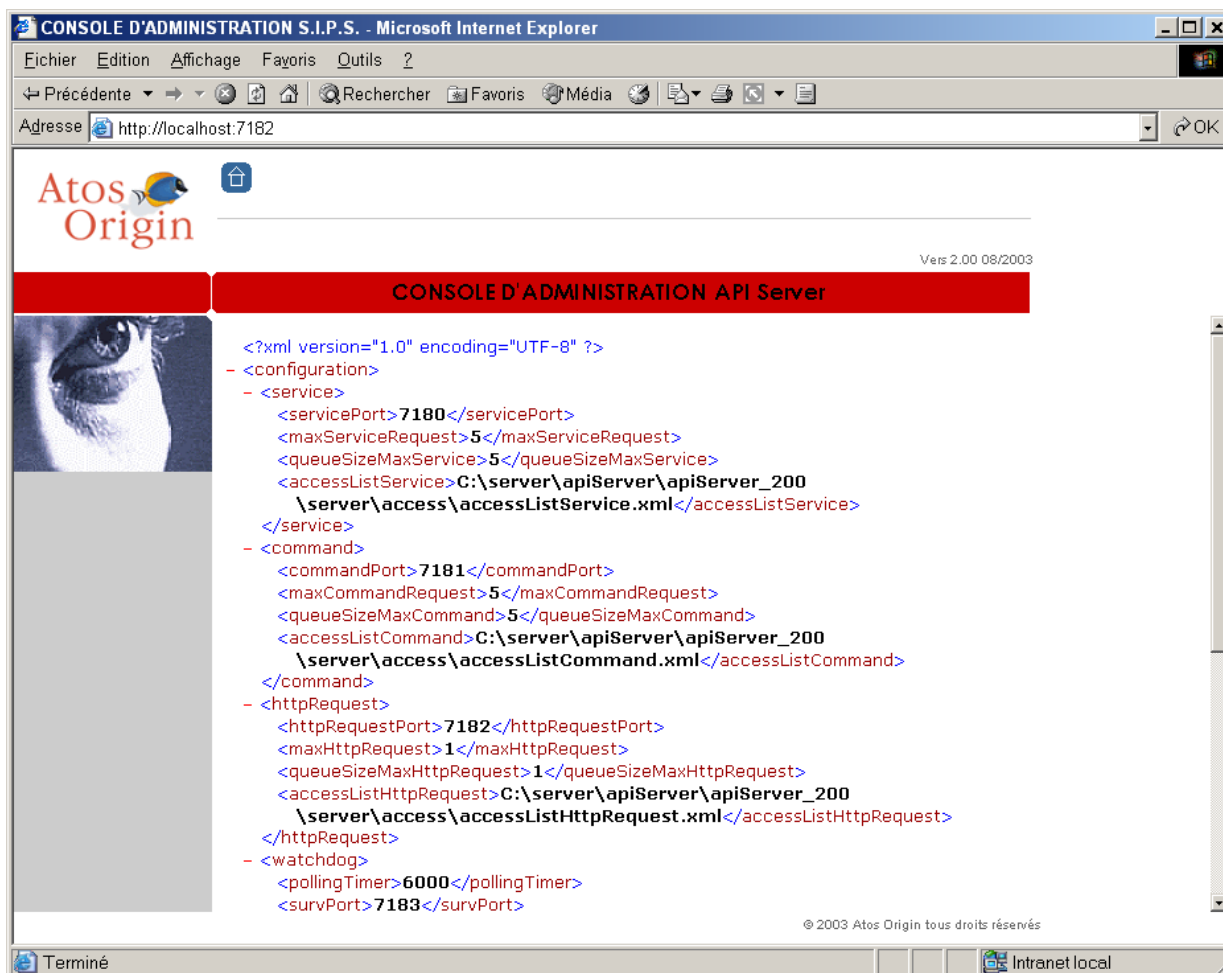
3.2.4 La commande « ENREGISTRER UN NOUVEAU COMPOSANT »

Dans le cas où vous ajoutez un composant à l'API Sips Office Server, vous devez utiliser cette commande pour mettre à jour les nouveaux services.



3.2.5 La commande « VOIR LA CONFIGURATION DU SERVEUR »

Cette fonction vous permet de visualiser le fichier de configuration config.xml via le navigateur.



3.2.6 La commande « CHANGER LE MOT DE PASSE D'ADMINISTRATION »

Cette fonction permet de changer le login et le mot de passe à travers l'accueil Http.

Note

Aucun contrôle de complexité n'est effectué sur le mot de passe. La gestion de ce mot de passe est laissée à l'administrateur.

CONSOLE D'ADMINISTRATION S.I.P.S. - Microsoft Internet Explorer

Fichier Edition Affichage Favoris Outils ?

Précédente Recherche Favoris Média

Adresse http://localhost:7182 OK

Atos Origin

Vers 2.00 08/2003

CONSOLE D'ADMINISTRATION API Server

Saisir le nouveau login

Saisir le nouveau mot de passe

Confirmer le nouveau mot de passe

Valider Retour

© 2003 Atos Origin tous droits réservés

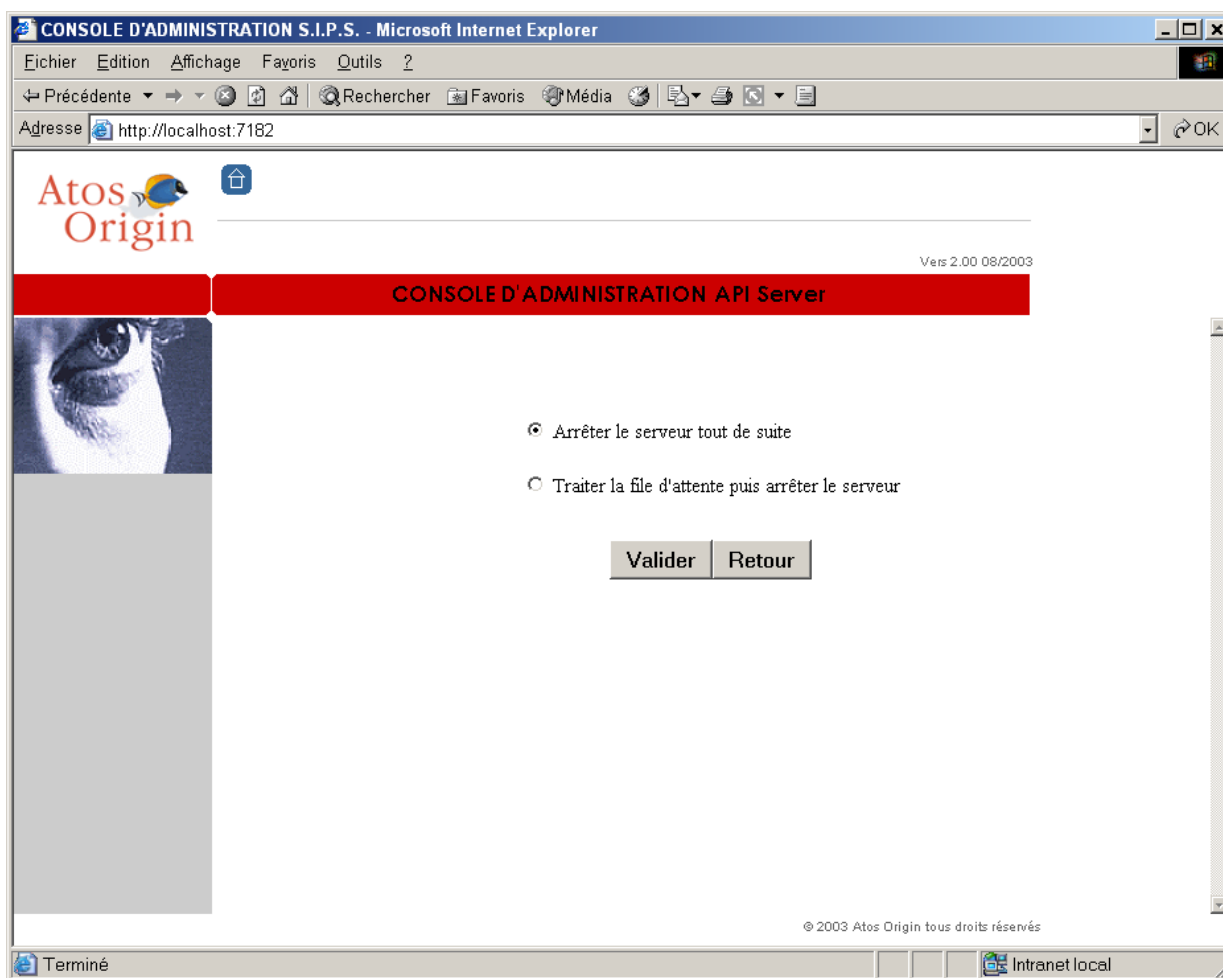
Terminé Intranet local

3.2.7 La commande « ARRETER LE SERVEUR »

Cette fonction vous permet de stopper le serveur de 2 manières différentes :

Soit stopper immédiatement, et ne pas traiter les requêtes en attente.

Soit arrêter, mais traiter auparavant les requêtes restantes dans la file d'attente.



4. ADMINISTRER EN MODE BATCH

La console d'administration n'est pas indispensable. En effet, elle utilise les mêmes principes que les autres requêtes de services. Par conséquent, les commandes essentielles peuvent être envoyées directement à l'API Sips Office Server sans utiliser la console, mais en envoyant directement une requête XML sur le port de commande que vous avez paramétré lors de l'installation.

Ceci peut s'avérer utile si on souhaite administrer l'API Sips Office Server en mode batch sans utiliser la console d'administration HTML.

Les requêtes et les réponses à envoyer vers le serveur d'accueil ont toutes la même structure. Ces structures sont décrites dans le *GUIDE DU DEVELOPPEUR*.

Note

Dans les exemples, toutes les trames XML sont présentées sur plusieurs lignes dans un souci de lisibilité, mais les trames reçues et renvoyées par l'API Sips Office Server seront toujours écrites sur une seule ligne (sans retour chariot).

4.1 SECURITE

Le problème levé par ce mode d'administration est la sécurité de l'API Sips Office Server. En effet, la console d'administration demande pour toute intervention l'authentification de l'utilisateur par login/password. Dans le cas de l'administration en mode batch, l'authentification n'est plus nécessaire que pour arrêter l'API Sips Office Server. Vous constaterez en effet l'absence des champs login et password dans chacune des trames XML décrites ci-après (sauf pour l'arrêt). La gestion de ce login/password est laissée à votre convenance. Rien ne vous empêche par exemple en créant votre batch d'arrêt de demander à l'utilisateur d'en saisir un.

4.2 LA COMMANDE « VERIFIER L'ETAT DU SERVEUR »

Cette commande vous permet de récupérer diverses informations sur l'état de l'API Sips Office Server. Elle suppose que l'API Sips Office Server soit lancé, sinon on obtient une erreur à l'initialisation de la connexion socket.

4.2.1 Paramètres de la requête

La requête de vérification de l'état du serveur ne contient pas de paramètres. Elle n'est constituée que de l'élément **command**.

Element command :

Attribut	Description	Valeur
component	Nom du composant	« status »
name	Nom de la commande	« getStatus »

4.2.2 Paramètres de la réponse

La réponse explicitant l'état du serveur est un élément **response** contenant le sous élément **getStatus**.

Element response :

Attribut	Description	Valeur
component	Nom du composant	« status »
name	Nom de la commande	« getStatus »

Element response/getStatus :

Attribut	Description	Valeur
startDate	Date de lancement de l'API Sips Office Server	AAAA/MM/JJ
startTime	Heure de lancement de l'API Sips Office Server	HH:MM:SS
serverDate	Date locale de l'API Sips Office Server	AAAA/MM/JJ
serverTime	Heure locale de l'API Sips Office Server	HH:MM:SS
serviceCount	Nombre de requêtes de service traitées	numérique
commandCount	Nombre de requêtes de commande traitées	numérique
errorCount	Nombre d'erreurs survenues	numérique
fatalCount	Nombre d'erreurs fatales survenues	numérique
startDate	Date de lancement de l'API Sips Office Server	AAAA/MM/JJ
startTime	Heure de lancement de l'API Sips Office Server	HH:MM:SS

L'élément **getStatus** contient en outre deux sous-éléments **lastRequest** et **lastCommand** décrivant les dernières requêtes traitées par l'API Sips Office Server.

Element response/getStatus/lastRequest :

Attribut	Description	Format
Date	Date de la dernière requête de service	AAAA/MM/JJ
Time	Heure de la dernière requête de service	HH:MM:SS
Name	Nature de la dernière requête de service	composant/fonction

Element response/getStatus/lastCommand :

Attribut	Description	Format
Date	Date de la dernière requête de commande	AAAA/MM/JJ
Time	Heure de la dernière requête de commande	HH:MM:SS
Name	Nature de la dernière requête de commande	composant/fonction

4.2.3 Trame de requête XML

```
<command component="status" name="getStatus"></command>
```

4.2.4 Exemple de trame de réponse XML

```
<response component="status" name="getStatus">
<getStatus startDate="2003/08/20"
startTime="11:11:35"
serverDate="2003/08/20"
serverTime="11:36:49"
serviceCount="0"
commandCount="3"
errorCount="0"
fatalCount="0" >
<lastRequest Date="null"
Time="null"
Name="null" >
</lastRequest>
<lastCommand Date="2003/08/20"
Time="11:13:29"
Name="password / change" >
</lastCommand>
</getStatus>
</response>
```

4.3 LA COMMANDE « LIRE LES TRACES »

Cette fonction vous permet d'effectuer une recherche multicritère dans les fichiers de traces du serveur d'accueil ou des composants.

L'« Heure de début » et l'« Heure de fin » permettent de réduire la lecture des traces sur un intervalle de chaque journée de la période définie par « Date de début » et « Date de fin ». Au contraire de l'interface d'administration, tous les champs sont obligatoires.

4.3.1 Paramètres de la requête

La requête de lecture des traces contient des paramètres obligatoires. Elle est constituée de l'élément **command** contenant le sous élément **getTraces**.

Element command :

Attribut	Description	Valeur
component	Nom du composant	« traces »
name	Nom de la commande	« getTraces »

Element command/getTraces :

Attribut	Description	Valeur ou Format
pattern	Nom des fichiers de traces à lire	valeur de l'attribut «prefix» du fichier config.xml
fromDate	date de début des traces à extraire	AAAAMMJJ
toDate	date de fin des traces à extraire	AAAAMMJJ
fromTime	Heure de début des traces à extraire	HHMMSS
toTime	Heure de fin des traces à extraire	HHMMSS

4.3.2 Paramètres de la réponse

La réponse de la lecture des traces n'est pas formatée en XML. On obtient une extraction des traces telles qu'elles sont enregistrées dans les fichiers de traces.

4.3.3 Trame de requête XML

```
<command component="traces" name="getTraces" >
<getTraces pattern="API Server"
fromDate="20030819"
toDate="20030820"
fromTime="150000"
toTime="114000" />
</command>
```

4.3.4 Exemple de réponse

```
232:11:11:34-0-(0)-IPServer : service server listening on port 7180
232:11:11:35-0-(0)-IPServer : command server listening on port 7181
232:11:11:36-0-(0)-RequestThread : -----
232:11:11:36-0-(0)-RequestThread : Type : command Component name : register Function : refresh
23EOT
```

4.4 LA COMMANDE « LISTER LES COMPOSANTS »

Vous pouvez lister les différents composants installés sur l'API Sips Office Server, leurs versions et les fonctionnalités qu'ils présentent.

4.4.1 Paramètres de la requête

Comme la requête `getStatus`, la requête de liste des composants ne présente pas de paramètres. Elle ne contient donc que l'élément **command**.

Element command :

Attribut	Description	Valeur
component	Nom du composant	« register »
name	Nom de la commande	« listComponents »

4.4.2 Paramètres de la réponse

La réponse explicitant l'état du serveur est un élément **response** qui rappelle la fonction demandée, et contient un élément **listComponents** qui décrit tous les composants installés.

Element response :

Attribut	Description	Valeur
component	Nom du composant	« register »
name	Nom de la commande	« listComponents »

Element response/listComponents:

L'élément **listComponents** contient un élément **APIItem** par composant de service ou de commande installé sur l'API Sips Office Server. Il présentera donc toujours au minimum quatre éléments **APIItem** correspondant aux quatre composants de commande installés par défaut avec l'API Sips Office Server.

Element response/listComponents/APIItem :

Attribut	Description	Valeur/Format
name	Nom du composant	an
version	Version du composant	an
creationDate	Date de création du composant	JJ/MM/AAAA
shortDesc	Description rapide	ans

L'élément **APIItem** contient deux éléments supplémentaires. L'élément **classId** décrit la classe java principale du composant, et l'élément **serviceList** qui contient un élément **service** par fonctionnalité possible du composant.

Element response/listComponents/APIItem/classID :

Attribut	Description	Valeur/Format
Name	Identifiant du composant	an
Class	nom de la classe principale du composant	com.atosorigin...

Element response/listComponents/APIItem/serviceList/service :

Attribut	Description	Valeur
name	Nom du service	an
fonction	Fonction rendant le service	an

4.4.3 Trame de requête XML

```
<command component="register" name="listComponents"></command>
```

4.4.4 Exemple de réponse

```
<response component="register" name="listComponents">
<listComponents>
<APIItem name="password" version="2.00" creationDate="15/08/2003" shortDesc="Change the
password" >
<classId name="password"
class="com.atosorigin.services.cad.apiserver.components.command.password.Password" />
<serviceList>
<service name="change" fonction="changePassword" />
</serviceList>
</APIItem>
<APIItem name="register" version="2.00" creationDate="16/06/2003" shortDesc="Used to refresh the
dtd's and the routing buffer" >
<classId name="register"
class="com.atosorigin.services.cad.apiserver.components.command.register.Register" />
<serviceList>
<service name="refresh" fonction="refresh" />
<service name="listComponents" fonction="listComponents" />
</serviceList>
</APIItem>
<APIItem name="status" version="2.00" creationDate="14/07/2003" shortDesc="Api Server status
command" >
<classId name="status"
class="com.atosorigin.services.cad.apiserver.components.command.status.StatusWrapper" />
<serviceList>
<service name="getStatus" fonction="getStatus" />
<service name="trace" fonction="traceStatus" />
</serviceList>
</APIItem>
<APIItem name="traces" version="2.00" creationDate="04/07/2003" shortDesc="Api Server traces
reader command" >
<classId name="traces"
class="com.atosorigin.services.cad.apiserver.components.command.traces.Traces" />
<serviceList>
<service name="getTraces" fonction="getTraces" />
</serviceList>
</APIItem>
</listComponents>
</response>
```

4.5 LA COMMANDE « ENREGISTRER UN NOUVEAU COMPOSANT »

Cette commande sert exclusivement dans le cas de l'ajout d'un nouveau composant à l'API Sips Office Server. Dans le cas d'une mise à jour, il faudra effectuer obligatoirement un arrêt/relance de l'API Sips Office Server.

4.5.1 Paramètres de la requête

La requête d'enregistrement de nouveaux composants ne contient pas de paramètres. Elle n'est constituée que de l'élément **command**.

Element command :

Attribut	Description	Valeur
component	Nom du composant	« register »
name	Nom de la commande	« refresh »

4.5.2 Paramètres de la réponse

La réponse de l'enregistrement de nouveaux composants est un élément **response** contenant le sous élément **refresh**.

Element response :

Attribut	Description	Valeur
component	Nom du composant	« register »
name	Nom de la commande	« refresh »

Element response/refresh :

Attribut	Description	Valeur
message	message de réponse	Mise à jour OK : 5 composants ont été détectés

Le nombre de composants détectés correspond au nombre de composants installés sur l'API Sips Office Server.

4.5.3 Trame de requête XML

```
<command component="register" name="refresh">
</command>
```

4.5.4 Exemple de réponse

```
<response component="register" name="refresh">
<refresh message="Mise à jour OK, 6 composants ont été détectés." />
</response>
```

4.6 LA COMMANDE « CHANGER LE MOT DE PASSE D'ADMINISTRATION »

Cette fonction permet de modifier le user et le mot de passe d'administration requis pour utiliser la console d'administration HTML et envoyer une requête d'arrêt vers l'API Sips Office Server.

4.6.1 Paramètres de la requête

La requête est un élément command contenant un sous-élément change. Les paramètres sont ici tous obligatoires.

Element command :

Attribut	Description	Valeur
component	Nom du composant	« password »
name	Nom de la commande	« change »

Element command/change :

Attribut	Description	Format
login	Nom du composant	ans
password	Nom de la commande	ans

Note

Aucun contrôle de complexité n'est effectué sur le mot de passe. La gestion de ce mot de passe est laissée à l'administrateur.

4.6.2 Paramètres de la réponse

La réponse du changement du user/mot de passe est un élément réponse contenant un sous-élément change.

Element response :

Attribut	Description	Valeur
component	Nom du composant	« password »
name	Nom de la commande	« change »

Element response/change :

Attribut	Description	Valeur
message	message de réponse	"Login and password are changed"

4.6.3 Trame de requête XML

```
<command component="password" name="change" >
<change login="new_login" password="new_password" />
</command>
```

4.6.4 Trame de réponse XML

```
<response component="password" name="change">
<change message="Login and password are changed" />
</response>
```

4.7 LA COMMANDE « ARRETER LE SERVEUR »

Cette fonction vous permet de stopper le serveur de 2 manières différentes :

- Soit stopper immédiatement, et ne pas traiter les requêtes en attente.
- Soit arrêter, mais traiter auparavant les requêtes restantes dans la file d'attente.

Note

- Cette requête arrête les deux serveurs watchdog et serveur d'accueil à la fois.
- Cette requête est traitée par le serveur d'accueil. Ce dernier doit donc être lancé pour pouvoir traiter cette commande correctement.

4.7.1 Paramètres de la requête

Cette requête ne contient pas d'attribut component. En effet, la commande d'arrêt n'est pas envoyée à un composant en particulier, mais à l'API Sips Office Server lui-même.

Note

Du fait de l'absence des attributs components et name, cette requête a un format différent de toutes les autres requêtes traitées par l'API Sips Office Server.

Les paramètres de la requête sont décrits ci-dessous :

Element command :

Attribut	Description	Valeur/Format
login	user d'administration	ans
password	password d'administration	ans
name	Nom de la fonction et son paramètre	arrêt immédiat : "StopServer EMPTY_QUEUE" arrêt après traitement de la queue des messages : "StopServer NOW"

4.7.2 Paramètres de la réponse

Cette fonction ne rend bien sûr pas de réponse puisque le serveur API Sips Office Server est alors arrêté.

4.7.3 Trames de requête XML

Pour arrêter le serveur après traitement de la file d'attente :

```
<command login="admin" password="admin" name="StopServer EMPTY_QUEUE">
</command>
```

Pour arrêter le serveur immédiatement.

```
<command login="admin" password="admin" name="StopServer NOW">
</command>
```


4.8 ABSENCE DE REPONSE ET MESSAGES D'ERREUR

Si vous ne recevez pas de réponse, vous devez vérifier les points suivants :

- L'API Sips Office Server est-il bien démarré ?
- N'y a-t-il pas un firewall entre le client et l'API Sips Office Server qui pourrait empêcher l'accès au serveur ?
- L'adresse IP de l'API Sips Office Server que vous utilisez pour votre connexion socket est-elle correcte ?
- Le port de l'API Sips Office Server que vous utilisez pour votre connexion socket est-il identique à celui configuré dans le fichier config.xml (cf. GUIDE D'ADMINISTRATION)
- La machine qui héberge votre client est-elle bien autorisée à accéder aux requêtes de commande dans le fichier accessListCommand.xml.
- Le composant que vous appelez est-il bien installé ?

Si vous obtenez un message d'erreur tel que celui présenté ci-dessous :

<Error message=" message d'erreur "/>

Cela signifie que la requête a bien été reçue par l'API Sips Office Server, mais elle comporte une erreur. Vous trouverez dans le tableau ci-dessous les principaux messages renvoyés par les composants ainsi que leur origine.

Exemple de messages	Cause	solution
"statut is not a registered component"	statut n'est pas un nom de composant enregistré	Corriger le nom de composant en vous référant aux exemples des paragraphes 4.2 à 4.7.
Attribut "name" is not valid.	la valeur de l'attribut name n'est pas un nom d'opération correct dans la requête XML.	Corriger le nom de fonction en vous référant aux exemples des paragraphes 4.2 à 4.7.
Attribute "toto" is not declared for element "getTraces".	Vous utilisez un nom d'attribut inconnu dans la requête XML.	Vérifier le nom des attributs en vous référant aux exemples des paragraphes 4.2 à 4.7.
URI=null Ligne = 15 : Root element type is "command", but was declared to be "service".	Vous utiliser le port de service de l'API Sips Office Server au lieu du port de commande	Utiliser le port de commande configuré dans le fichier config.xml pour votre connexion socket à l'API Sips Office Server (cf. (cf. <i>GUIDE D'ADMINISTRATION</i>).
Autres messages		Contacteur le Centre d'Assistance Technique

5. DESCRIPTION DES TRACES

5.1 FORMAT DES TRACES

Toutes les traces sont écrites avec le même format.

JJJ:HH:MM:SS-(ID)-N-message

Les données de ce format sont les suivantes :

Donnée	Description
JJJ	Le numéro de jour dans l'année. Il peut être sur une, deux ou trois positions.
HH	L'heure où la trace a été écrite
MM	la minute où la trace a été écrite
SS	La seconde où la trace a été écrite
ID	Identifiant du processus, numérique sur 4 caractères maximum.
N	Le niveau de la trace. Les niveaux possibles sont 0 : trace de fonctionnement normal 1 : trace d'erreur fatale 2 : trace d'erreur 3 : trace warning 4 : trace debug
message	Le message de trace.

5.2 NIVEAUX DE TRACES

Le niveau des traces est paramétré dans le fichier config.xml, il s'agit de l'attribut *level* du sous-élément *trace* de l'élément *watchdog*. La prise en compte de la modification de cette valeur nécessite un arrêt/relance de l'API Sips Office Server.

Niveau NORMAL (0) :

Un niveau de trace normal ne fera apparaître que les événements standards tels que le démarrage du serveur ou la réception d'une requête.

Niveau FATAL (1) :

Les traces de niveau fatal correspondent aux cas où le serveur refuse de démarrer, et aux cas où le serveur de polling détecte que le serveur d'accueil est tombé.

Niveau ERROR (2) :

Ce niveau comprend toutes les erreurs qui empêchent le traitement des requêtes. Par exemple, mauvais format de paramètre, IP non autorisée.

Niveau WARNING (3) :

On trace dans ce niveau les erreurs liées au développement. Par exemple requête mal formée, erreur sur l'attribut, erreur sur le login/mot de passe d'administration.

Niveau DEBUG (4) :

Ce niveau fera apparaître tous les autres niveaux ainsi qu'une description complète des phases de traitement d'une requête.

Note

- Dans les niveaux 3 et 4, les fichiers de traces deviennent rapidement volumineux, et chaque ligne de traces correspond à un appel RMI. Ces niveaux ne sont par conséquent pas adaptés à une utilisation en production.
- Les niveaux 0, 1 et 2 ne peuvent être masqués. Par conséquent paramétrer level=0, 1 ou 2 dans le fichier config.xml produira toujours les mêmes traces.

5.3 TRACES DE L'API SIPS OFFICE SERVER

La description des traces des composants de service est faite dans les GUIDES DU COMPOSANT correspondant.

Les traces du serveur d'accueil (correspondant à l'élément *trace* du fichier config.xml) contiennent à la fois les traces du serveur lui-même et de tous les composants de commande.

Vous trouverez ci-dessous un exemple de trace de démarrage du serveur suivi d'une commande listComponents.

```
206:11:31:41-0-(0)-IPServer : service server listening on port 7180
206:11:31:41-0-(0)-IPServer : command server listening on port 7181
206:11:32:00-0-(0)-RequestThread : -----
206:11:32:00-0-(0)-RequestThread : Type : command Component name : status Function : getStatus
206:11:32:00-0-(1)-RequestThread : -----
206:11:32:00-0-(1)-RequestThread : Type : command Component name : register Function :
listComponents
```

Cet exemple correspond au lancement de l'API Sips Office Server avec un niveau de traces 0. Les niveaux 0, 1 et 2 ne sont pas masquables, vous pourrez par exemple obtenir les traces suivantes dans le cas où la queue des requêtes est saturée :

```
205:16:08:22-2-(2)-IPServer : Request denied : command
```

Si vous démarrez l'API Sips Office Server avec le niveau de traces Warning (3), vous obtiendrez des traces supplémentaires. Par exemple si un composant n'est pas enregistré :

```
206:16:38:30-3-(0)-RequestThread : Request : <command component="statut"
name="getStatus"></command>
```

```
Response : <Error message="statut is not a registered component"/>
```

Enfin en niveau Debug (4), tout le traitement des requêtes est détaillé dans les traces, ainsi :

```
206:16:38:25-4-(0)-IPServer : Initialization of the status
206:16:38:25-4-(0)-AccessVerification : AccessList :
C:\server\Apig\server\access\accessListHttpRequest.xml
206:16:38:30-4-(0)-IPServer : Address : 127.0.0.1 accepted
206:16:38:30-4-(0)-IPServer : component name : statut
```

6. REMONTEE D'ALERTE

Au cours du fonctionnement du serveur, il peut être nécessaire de faire remonter automatiquement des alertes vers une application de surveillance déjà en fonction sur la plate-forme technique du commerçant.

Le système de remontée d'alerte de l'API Sips Office Server est basé sur une interface HTTP : lorsque qu'un événement survient sur l'API Sips Office Server, celui ci génère une requête POST http vers l'URL paramétrée dans le fichier config.xml et envoie le message d'erreur correspondant.

Les alertes sont remontées selon ce principe uniquement lorsqu'une trace de type FATAL est inscrite.

Le message d'erreur est envoyé dans une variable de nom « errorMsg ».

Les cas possibles de ce message sont :

- **Watchdog dead or not responding** : le process watchdog ne répond plus
- **Server dead or echo port not responding** : le serveur d'accueil ne répond plus
- **BootThread : "message exception java"** : le process qui traite une requête n'a pu être initialisé, la cause est précisée dans le message d'exception.