

Sips Payment Web JAVA 6.15

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

Sommaire

1. INTRODUCTION.....	2
2. LISTE DES OBJETS LIVRES	3
3. SCHEMA DES FLUX ET APPEL DES SERVLETS.....	4
3.1 POUR UN PAIEMENT NON 3-D SECURE	4
3.2 POUR UN PAIEMENT 3-D SECURE AVEC UNE CARTE ENROLEE	5
4. ETAPES D'INSTALLATION POUR EXECUTER LES EXEMPLES SUR LE SERVEUR DE DEMONSTRATION	6
4.1 SI VOUS ETES INSCRIT AU PROGRAMME 3-D SECURE	6
4.2 APPEL DU SERVEUR SIPS PAIEMENT	7
4.3 PROCEDURE DE TEST SUR LE SERVEUR SIPS DE DEMONSTRATION	8
4.3.1 <i>Test de paiement pour un commerçant non inscrit au programme 3-D Secure</i>	8
4.3.2 <i>Test de paiement pour un commerçant inscrit au programme 3-D Secure</i>	9
4.4 RECEVOIR LA REPONSE A LA FIN D'UN PAIEMENT	9
4.5 RECEVOIR LA REPONSE AUTOMATIQUE DU SERVEUR SIPS.....	10
5. DEVELOPPEMENT DE VOS SERVLETS	11
6. COMMENT PASSER SUR LE SERVEUR DE PRODUCTION	12

1. INTRODUCTION

Ce document vous explique comment installer votre API Sips Payment Web et comment démarrer vos premiers tests de paiement. Pour installer l'API, vous devrez développer 3 Servlets et configurer 3 fichiers de paramètres. La mise en place complète de l'API avant le passage en production se déroule en 3 phases :

- les premiers tests avec les programmes d'exemples fournis pour comprendre le fonctionnement de l'API,
- le développement de vos Servlets pour l'intégration à votre site,
- le passage en « pré-production » pour tester une demande d'autorisation, et l'installation éventuelle de votre charte graphique.

Note : ce document ne décrit pas comment vous interfacer avec votre système d'information ou votre base de données. Dans les exemples fournis, les variables sont déjà renseignées, vous devrez programmer la lecture et la mise à jour des données de votre système d'information.

Pré-requis :

- Connaissances de base du langage Java.
- Connaissances de base de l'écriture et de la compilation des Servlets.
- Compilateur Java pour construire vos Servlets.
- Moteur de Servlets.

2. LISTE DES OBJETS LIVRES

L'API est livrée sous la forme d'un fichier .tar contenant les fichiers suivants :

Fichier /Version.txt

fichier précisant l'environnement dans lequel l'API a été compilée et testée.

Répertoire /jar

apipayment.jar

Archive Java des classes de l'API Sips Payment Web.

Répertoire /logo

répertoire des logos des moyens de paiement

Répertoire /param

certif.fr.011223344551111.jsp

Certificat de la boutique de démonstration

certif.fr.011223344551112.jsp

Certificat de la boutique de démonstration 3D

parmcom.011223344551111

Fichier des paramètres de la boutique de démonstration

parmcom.defaut

Fichier des paramètres par défaut

Pathfile

Fichier des chemins d'accès aux fichiers paramètres

Répertoire /source

RequestServlet.java

exemple d'appel Sips Payment Web

ResponseServlet.java

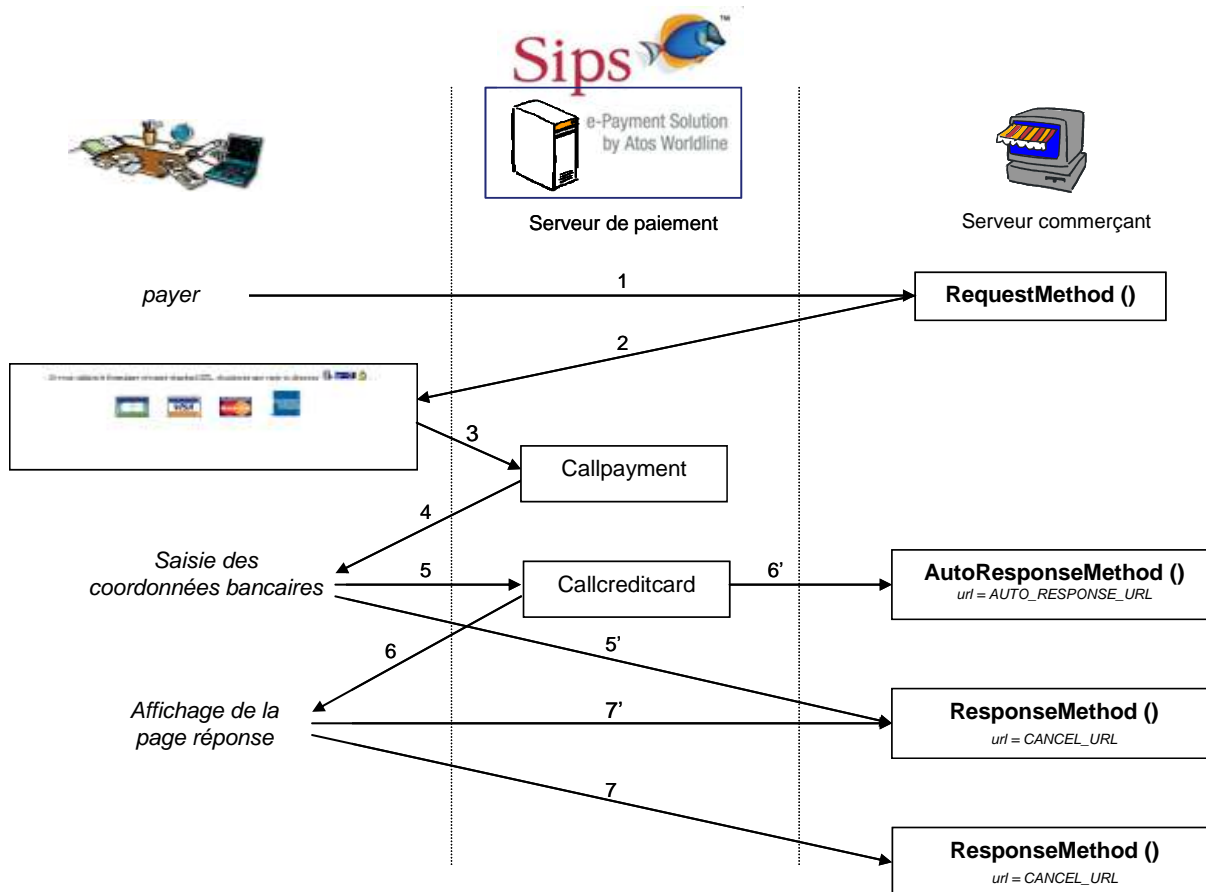
exemple de réponse Sips Payment Web

AutoResponseServlet.java

exemple de réponse automatique Sips Payment Web

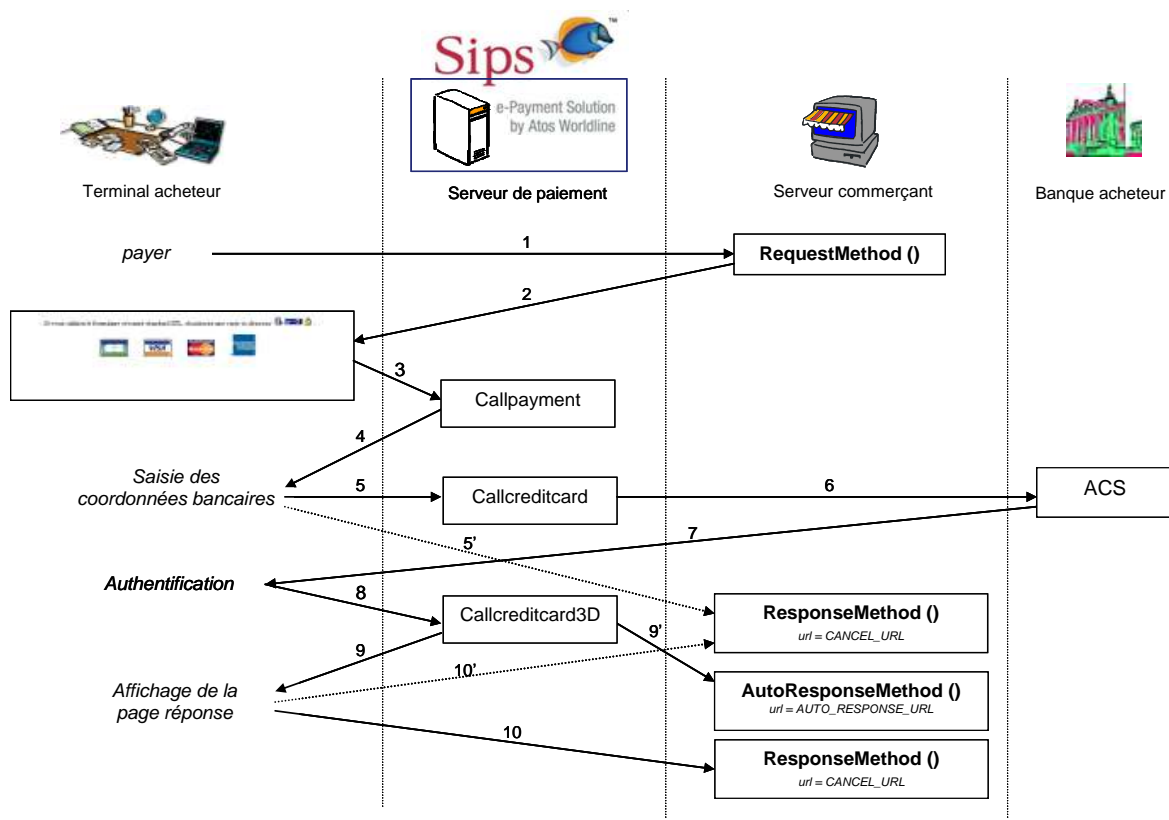
3. SCHEMA DES FLUX ET APPEL DES SERVLETS

3.1 POUR UN PAIEMENT NON 3-D SECURE



- 1 L'internaute a rempli son caddie, il souhaite passer à la caisse
- 2 La méthode « RequestMethod » est exécutée, elle affiche tous les moyens de paiement acceptés par la boutique.
- 3 En cliquant sur un de ces moyens de paiement, l'internaute se connecte au serveur de paiement.
- 4 Le serveur de paiement envoie le formulaire approprié pour obtenir ses coordonnées bancaires.
- 5 Après saisie de ses coordonnées bancaires, l'internaute soumet le formulaire au serveur de paiement qui va effectuer une demande d'autorisation auprès d'une institution financière impliquée dans le paiement sélectionné.
- 5' L'internaute peut ne pas saisir ses coordonnées bancaires et annuler sa transaction.
- 6 et 6' Après réception de la réponse d'autorisation, le serveur de paiement envoie simultanément une réponse au serveur commerçant (appel de la méthode « AutoresponseMethod ») ainsi qu'un ticket de caisse sur l'écran de l'internaute.
- 7 et 7' A partir de ce ticket, l'internaute peut cliquer sur un bouton pour retourner sur le serveur commerçant, qui reçoit ainsi le résultat du paiement dans une variable cachée. La méthode « ResponseMethod » récupère le résultat et affiche une page appropriée.
Si le paiement est refusé, l'internaute suit le flux 7'.

3.2 POUR UN PAIEMENT 3-D SECURE AVEC UNE CARTE ENROLEE



- 1 L'internaute a rempli son caddie, il souhaite passer à la caisse
- 2 La méthode « RequestMethod » est exécutée, elle affiche tous les moyens de paiement acceptés par la boutique.
- 3 En cliquant sur un de ces moyens de paiement, l'internaute se connecte au serveur de paiement.
- 4 Le serveur de paiement envoie le formulaire approprié pour obtenir ses coordonnées bancaires.
- 5 Après saisie de ses coordonnées carte, l'internaute soumet le formulaire au serveur de paiement.
- 5' L'internaute peut ne pas saisir ses coordonnées bancaires et annuler sa transaction.
- 6 Le serveur de paiement transfère les coordonnées carte vers la banque du porteur.
- 7 La banque du porteur affiche un formulaire pour saisie du mot de passe et vérifie le mot de passe saisi.
- 8 Le serveur de paiement est destinataire de la vérification du mot de passe, et envoie une demande d'autorisation vers le réseau bancaire.
- 9 et 9' Après réception de la réponse d'autorisation, le serveur de paiement envoie simultanément une réponse au serveur commerçant (appel de la méthode « AutoresponseMethod») ainsi qu'un ticket de caisse sur l'écran de l'internaute.
- 10 et 10' A partir de ce ticket, l'internaute peut cliquer sur un bouton pour retourner sur le serveur commerçant, qui reçoit ainsi le résultat du paiement dans une variable cachée. La méthode « RequestMethod » récupère le résultat et affiche une page appropriée.
Si le paiement est refusé, l'internaute suit le flux 10'.

4. ETAPES D'INSTALLATION POUR EXECUTER LES EXEMPLES SUR LE SERVEUR DE DEMONSTRATION

Cette première phase a pour but d'effectuer votre premier paiement sur le serveur de démonstration. Les 3 exemples fournis vous permettent de traiter les 3 messages échangés entre votre serveur, le navigateur de l'acheteur et le serveur de paiement.

Commencez par détarer le fichier Sips_XXX_Java_XXXX.tar en utilisant la commande suivante :

```
tar -xvf Sips_XXX_Java_XXXX.tar
```

ou Winzip si vous êtes sous Windows.

Vous obtiendrez dans les deux cas l'ensemble des fichiers décrits au paragraphe 2.

Voici maintenant quatre étapes pour simuler votre premier paiement :

4.1 SI VOUS ETES INSCRIT AU PROGRAMME 3-D SECURE

Le commerçant 011223344551111 n'est pas inscrit par défaut au programme 3-D Secure. Pour effectuer des tests avec un commerçant de démonstration considéré comme « inscrit » au programme 3-D Secure, vous devez utiliser le commerçant 011223344551112.

Pour ceci effectuez au préalable les 2 opérations suivantes :

- dupliquer le fichier parmcom.011223344551111 et le renommer en parmcom.011223344551112.
- remplacer le numéro de la boutique de démonstration (011223344551111) par le numéro de la boutique de démonstration 3-D Secure (011223344551112) (cf. champ *merchant_id*) dans le fichier source RequestServlet.java.

4.2 APPEL DU SERVEUR SIPS PAIEMENT

1. Modifiez le contenu du pathfile en fonction de votre répertoire d'installation.

Avec un OS Windows, par exemple :

F_CERTIFICATE!c:\Sips\payment\param\certif!

F_PARAM!c:\Sips\payment\param\paramcom!

F_DEFAULT!c:\Sips\payment\param\paramcom.default !

Avec un OS de type Unix, par exemple :

F_CERTIFICATE!/home/Sips/payment\param/certif!

F_PARAM!/home/Sips/payment\param\paramcom!

F_DEFAULT!/home/Sips/payment\param\paramcom.default !

Notez que les fichiers F_CERTIFICATE et F_PARAM ne prennent pas d'extension (.011223344551111.jsp ou .011223344551112.jsp). Ces données seront rajoutées par l'API.

Des chemins complets sont recommandés, mais les chemins relatifs fonctionnent.

Le type de certificat est renseigné par : F_CTYPE !jsp!

Cela correspond à l'extension du fichier certificat suivant 011223344551111 dans le nom du certificat

Renseignez D_LOGO avec un chemin internet : Si vos logos sont accessibles depuis un navigateur sur l'URL http://www.maboutique.fr/logo_api/CB.gif,

il faudra renseigner : D_LOGO!/logo_api/!

Renseignez DEBUG à YES si vous souhaitez passer en mode DEBUG.

2. Editez **RequestServlet.java** avec un compilateur standard Java.
3. Modifiez le chemin du fichier **pathfile**. Il est recommandé d'indiquer un chemin complet, mais un chemin relatif fonctionnera également.
4. Vérifiez que le fichier **pathfile** a bien les droits de lecture pour la servlet **RequestServlet.java**.
5. Compilez. Votre projet Java doit bien sûr inclure apipayment.jar et servlet.jar, cette dernière librairie étant la librairie standard des Servlets Java, présente dans les moteurs de Servlets.
6. Installez la librairie Java apipayment.jar dans le répertoire des librairies de votre moteur de Servlets. Un arrêt/relance de ce dernier est nécessaire pour prendre en compte cette nouvelle librairie.
7. Copiez la Servlet compilée dans votre répertoire Servlet et appelez-la depuis un navigateur. Vous devez voir apparaître les logos des cartes bancaires.

4.3 PROCEDURE DE TEST SUR LE SERVEUR SIPS DE DEMONSTRATION

4.3.1 Test de paiement pour un commerçant non inscrit au programme 3-D Secure

Vous devez utiliser le commerçant 011223344551111

Sur le serveur de démonstration Sips, le processus d'autorisation est simulé. Il est donc possible de saisir n'importe quel numéro de carte sans aucune conséquence.

Le code réponse de la demande d'autorisation simulée (champ **bank_response_code**) correspond aux deux derniers chiffres du numéro de carte bancaire.

Le champ **response_code** est simulé selon la valeur du **bank_response_code**.

Exemple :	numéro de carte	code réponse
	4974934125497800	00 (paiement accepté)
	4972187615205	05 (paiement refusé)

Attention : la date de validité de la carte doit être postérieure à la date du jour. La taille du numéro de carte doit être comprise entre 10 et 19 chiffres.

Pour connaître les valeurs possibles des champs **bank_response_code** et **response_code** en production, référez-vous au *DICIONNAIRE DES DONNEES*.

Pour les cartes CB, VISA et MASTERCARD, vous devez saisir un cryptogramme visuel (clé sécuritaire à trois chiffres). Comme pour le numéro de carte, les 2 derniers chiffres simulent le **cvv_response_code**.

Exemple :	cryptogramme	cvv_response_code
	600	4D
	640	4D
	650	50
	653	53
	655	55

Tout cryptogramme dont les 2 derniers chiffres diffèrent de 00, 40, 50, 53 ou 55 conduit à un **cvv_response_code** égal à 4E.

Pour connaître la signification des différentes valeurs du **cvv_response_code**, reportez-vous à l'annexe traitant du cryptogramme visuel dans le *DICIONNAIRE DES DONNES*.

4.3.2 Test de paiement pour un commerçant inscrit au programme 3-D Secure

Vous devez utiliser le commerçant 011223344551112

Pour les demandes d'autorisation le processus de simulation (des champs **response_code**, **bank_response_code** et **cvv_response_code**) est le même que pour un commerçant non inscrit au programme 3-D Secure (cf. paragraphe précédent).

Les différents cas de traitement 3D sont simulés à partir du numéro de carte :

- simuler une carte VISA enrôlée : saisir un numéro de carte commençant par 4
- simuler une carte MASTERCARD enrôlée : saisir un numéro de carte commençant par 5
- simuler une erreur technique pendant le processus d'authentification : saisir un numéro de carte commençant par 9
- simuler un paiement avec une carte VISA ou MASTERCARD non enrôlée : saisir un numéro de carte ne commençant pas par 4, 5 ou 9
- simuler le dialogue avec l'ACS dans la langue anglaise : saisir un numéro de carte dont le deuxième chiffre est un 0

Si vous saisissez un numéro de carte enrôlé, le mot de passe pour s'authentifier est « 00000000 ».

4.4 RECEVOIR LA REPONSE A LA FIN D'UN PAIEMENT

Nous supposons que vous avez maintenant terminé les deux premières étapes, vous êtes donc maintenant capable de connecter votre internaute au serveur Sips en vue d'un paiement.

Afin de recevoir la réponse du serveur Sips sur votre serveur Web, il vous faut avoir configuré au préalable une adresse URL de réponse pour que le serveur Sips sache où il doit envoyer la réponse.

La manière la plus facile de le faire est de configurer l'URL de réponse dans le fichier paramètre (parmcom.011223344551111 ou parmcom.011223344551112).

1. Editez **ResponseServlet.java**
2. Modifiez le chemin du fichier **pathfile**.
3. Compilez avec un compilateur standard Java. Utilisez de préférence le même projet Java que précédemment, ce qui évite les soucis d'inclusion de bibliothèques.
4. Copiez la Servlet compilée dans le répertoire Servlet de votre moteur de Servlets.
5. Editez le fichier parmcom.011223344551111 (ou parmcom.011223344551112), recherchez les paramètres **RETURN_URL** et **CANCEL_URL**, et changez les valeurs pour appeler **ResponseServlet** sur votre site web.

exemple :

RETURN_URL!http://www.maboutique.fr/servlet/ResponseServlet!

CANCEL_URL!http://www.maboutique.fr/servlet/ResponseServlet!

6. Effectuez un nouveau paiement en utilisant la Servlet compilée dans la première étape. Sur la deuxième page de paiement, le bouton « RETOUR A LA BOUTIQUE » doit rediriger vers la Servlet **ResponseServlet** (vous pouvez le vérifier en regardant la source sur votre navigateur).
7. Lorsque vous cliquez sur le bouton « RETOUR A LA BOUTIQUE », vous quittez le serveur Sips et retournez sur le serveur commerçant via la Servlet **ResponseServlet**. Le programme livré en exemple affichera entièrement la structure de l'objet de réponse reçue par le serveur Sips.

4.5 RECEVOIR LA REPONSE AUTOMATIQUE DU SERVEUR SIPS

Vous avez effectué votre premier paiement sécurisé en ligne. Félicitations ! Comme vous l'avez probablement remarqué, la réponse a été renvoyée par le serveur de paiement Sips à votre serveur lorsque vous avez cliqué sur le bouton « RETOUR A LA BOUTIQUE ». Le risque avec ce mode d'envoi de réponse est que l'internaute ne clique pas sur ce bouton et que votre serveur ne reçoive jamais la réponse à la requête de paiement.

Afin d'éviter ce risque, une autre réponse a été configurée dans le serveur de paiement Sips, elle s'appelle la réponse automatique. C'est une réponse systématique envoyée par le serveur Sips à votre serveur commerçant à chaque fois qu'il traite une demande d'autorisation, donc à chaque fois qu'un internaute valide ses coordonnées carte.

Pour ce faire, le serveur Sips envoie une requête HTTP au serveur commerçant, avec l'URL spécifiée (paramètre `AUTO_RESPONSE_URL`).

Remarque : Notez qu'il s'agit ici de dialogue serveur/serveur. L'absence de navigateur interdit les redirections et les affichages. La servlet placée sur cette URL ne doit effectuer que des traitements de type mise à jour de base de données ou logging.

1. Editez **AutoResponseServlet.java**
2. Modifiez le chemin du fichier **pathfile**.
3. L'exemple livré copie le contenu de la structure de réponse dans un fichier de trace sur la machine qui exécute la servlet. Vous pouvez modifier le chemin du fichier trace dans la servlet.
4. Compilez votre projet avec un compilateur standard Java.
5. Copiez la servlet compilée dans le répertoire Servlet de votre moteur de Servlets.
6. Editez le fichier `parmcom.011223344551111` (ou `parmcom.011223344551112`), recherchez le paramètre `AUTO_RESPONSE_URL`, changez la valeur pour appeler la nouvelle servlet sur votre site et retirez le # si la ligne est en commentaire.

Exemple :

`AUTO_RESPONSE_URL!http://www.maboutique.fr/servlet/AutoResponseServlet!`

7. Effectuez un nouveau paiement, en utilisant la servlet compilée dans la première étape.
8. Lorsque vous soumettez votre paiement sur la « page de saisie », le serveur de paiement Sips appelle l'URL de réponse automatique et affiche la page de réponse, au même moment. Lorsque votre client lit cette seconde page, vous connaissez déjà le résultat du paiement.

Remarque : si vous ne recevez pas la réponse automatique, référez-vous au *GUIDE DU PROGRAMMEUR*.

5. DEVELOPPEMENT DE VOS SERVLETS

Au cours de cette phase, vous allez adapter les programmes d'exemples à vos besoins, pour qu'ils s'intègrent bien à votre serveur web et qu'ils échangent les données nécessaires avec votre système d'information.

Vous poursuivrez vos tests sur le serveur de « démonstration » qui affiche rigoureusement les mêmes pages que le serveur de « production », mais qui simule la demande d'autorisation à la banque. De cette manière vous pouvez multiplier les essais de paiements, et vous n'avez pas besoin d'utiliser une carte réelle.

Pour plus de détails sur les données de l'API, veuillez consulter le *GUIDE DU PROGRAMMEUR*.

6. COMMENT PASSER SUR LE SERVEUR DE PRODUCTION

La dernière phase est le passage en mode « pré-production ». A partir de ce moment les paiements sont effectués sur le serveur de production. Les demandes d'autorisation ne sont plus simulées comme sur le serveur de démonstration, mais elles empruntent le circuit complet du réseau bancaire. Cette phase va permettre de contrôler la bonne inscription de votre contrat bancaire et de tester la personnalisation de vos pages de paiement (cf. le *GUIDE DE PERSONNALISATION DES PAGES* et le *GUIDE DU PROGRAMMEUR*).

Le passage en « pré-production » se fait par l'activation du certificat du commerçant qui va remplacer le certificat de démonstration.

Tant que vous êtes en « pré-production » un message d'alerte s'affiche sur la page de saisie du numéro de carte, les paiements effectués sur votre site ne sont pas débités.

1. Déposer le certificat de « production », qui vous a été transmis, dans le même répertoire que le certificat de « démonstration ». Ce certificat est nommé `certif.fr.<my_merchant_id>.jsp` où `<my_merchant_id>` est votre numéro de boutique.
2. Renommer le fichier des paramètres de la boutique (fichier `parmcom.011223344551111` ou `parmcom.011223344551112`) en `parmcom.<my_merchant_id>`.
3. Remplacer le numéro de la boutique de démonstration (011223344551111 ou 011223344551112) par votre numéro de boutique (cf. champ *merchant_id*) dans vos Servlets.
4. Compilez vos Servlets et copier les Servlets modifiées dans le répertoire Servlet de votre moteur de Servlets.
5. Faire au minimum un test de paiement autorisé. Pour ce test, vous devez utiliser un numéro de carte réelle. Tant que vous êtes en phase de pré-production, les paiements effectués sur votre site ne sont pas débités. Vous ne pourrez passer en « production » que si au moins un paiement a été autorisé.
6. Si vous avez envoyé des logos ou des templates à installer sur le serveur Sips, vous pouvez les tester à ce moment en renseignant leurs noms dans les champs correspondants (cf. *DICTIONNAIRE DES DONNEES* et *GUIDE DE PERSONNALISATION DES PAGES*).
7. Pour le passage en « production », voir le document de *PRESENTATION FONCTIONNELLE* paragraphe **Comment en profiter**.