

# Module SIPS/ATOS pour Prestashop

Module et documentation écrits par Damien VERON.

<http://www.capillotracteur.fr>

module diffusé sous licence GNU/GPL v3.

Sont considérés comme faisant partie du module, et donc couverts par la GPL, les fichiers :

tgg\_atos/tgg\_atos.php

ainsi que le contenu des dossiers

tgg\_atos/tpl

tgg\_atos/admin-tpl

tgg\_atos/front-ctrl

ainsi que cette documentation.

Les autres fichiers ont qualité de ressources externes.

## Table des matières

Module SIPS/ATOS pour Prestashop.....	<u>1</u>
Installation :.....	<u>2</u>
Droits nécessaires sur les fichiers :.....	<u>3</u>
Généralités à propos de ce module.....	<u>4</u>
Adapter l'apparence du module à celle de votre boutique (optionnel).....	<u>4</u>
L'emplacement des fichiers de configuration (fs) est trop long (54 caractères maximum).....	<u>5</u>
Ajout d'une nouvelle devise au module :.....	<u>6</u>
Ajout d'une nouvelle banque au module.....	<u>7</u>
Passer le module en production.....	<u>8</u>
Questions fréquentes.....	<u>9</u>
Ma boutique est en maintenance et la commande n'apparait pas dans le back-office si je ne clique pas sur le bouton de retour à la boutique, mon module est-il mal installé ?.....	<u>9</u>
Le mode debug indique un code de retour 139 lors de l'appel à l'un des exécutable, qu'est-ce que cela signifie ?.....	<u>9</u>
Lorsque j'utilise le mode démonstration, mon numéro de carte bancaire est refusé, mon compte en banque est pourtant loin d'être en négatif, pourquoi ?.....	<u>9</u>
Développer autour de ce module.....	<u>10</u>
Hook tggAtosBankReturn.....	<u>10</u>
Hook tggAtosOrderConfirm.....	<u>11</u>

## **Installation :**

- Décompressez le dossier *tgg\_atos/* de l'archive dans le dossier *modules/* de votre installation de Prestashop.
- (facultatif mais très fortement conseillé) Obtenez les fichiers binaires ATOS fournis par votre banque correspondant à votre système d'exploitation, supprimez ceux du répertoire *bin/* puis placez ceux de votre banque. Obtenez le fichier *parmcom.<nomduservice>* fournis par votre banque pour remplacer celui présent dans le dossier *param*. Ceux fournis le sont pour vous permettre de vérifier rapidement que ce module répond à vos besoin. Ils devraient être remplacés avant un passage en production.
- Paramétrez les droits des fichiers (cf section appropriée de la documentation).
- Installez le module depuis le back-office de prestashop, cliquez sur **Configurer**. Si des erreurs sont affichées, réglez les paramètres en conséquence. Vérifiez au moins les options paramétrées sur la page de configuration basique.
- Depuis la page Paiement du back-office, pensez à paramétrer les devises pouvant être utilisées par le module (doivent être acceptées par votre banque).
- Idéalement, vous devriez déplacer les dossier */bin/*, */param/*, */log/* hors de la racine web de votre serveur puis ajuster les paramètres de la page de configuration du module en conséquence.
- Obtenir de votre banque le **Guide d'installation API Plug-in** de votre banque et le lire. Le nom peut différer selon les banques. Vous y trouverez les informations à propos du fonctionnement du système, les numéros de carte à utiliser lors du mode démonstration et des informations sur le passage en production du module.

## **Droits nécessaires sur les fichiers :**

Les droits seront expliquées en tant que différences par rapport aux droits génériques.  
Les fichiers `.htaccess` ne seront pas concernés lorsque l'on parlera du contenu d'un dossier.

- *droits par défaut :*

lecture et exécution pour les serveurs HTTP/PHP

- *tgg\_atos/*

lecture et exécution pour tous

- *tgg\_atos/images/* et son contenu
- *tgg\_atos/card\_logo/* et son contenu
- *tgg\_atos/logo.gif*
- *tgg\_atos/front-ctrl/* et son contenu

lecture et exécution par tous

- *tgg\_atos/log/*

- *tgg\_atos/param/*

- *tgg\_atos/fr.php* et autres fichiers de langue

tous les droits pour les serveurs HTTP/PHP

- *le contenu de tgg\_atos/bin/*

doit avoir des droits suffisants pour être exécuté via la fonction `exec()` en PHP

- *fichiers .htaccess*

lecture par tous

La tarball contient les droits utilisés lors du développement sur une offre mutualisée pro OVH, si vous souhaitez les conserver extrayez directement son contenu sur le serveur (`tar -xzf nomdufichier.tar.gz`).

## Généralités à propos de ce module

- Développé pour la branche 1.3 de Prestashop, rétrocompatibilité 1.2.5.
- Les erreurs générées sur le front office vous sont reportées par mail.
- Ce module gère le multilinguisme et transmet au serveur de paiement le code ISO de la langue utilisée par le visiteur. Attention à ce que le serveur bancaire accepte tous les codes ISO des langues actives sur le front office de votre boutique.
- Étant donné le caractère sensible d'une liaison bancaire, ce module doit être installé par une personne ayant les connaissances nécessaires à l'administration d'un serveur d'hébergement. Pour la même raison, ne sera fourni de support quant à l'installation du module qu'à des personnes ayant le verni technique nécessaire.

### Adapter l'apparence du module à celle de votre boutique (optionnel).

Soit *votre\_theme* le nom du thème utilisé (par défaut: *prestashop*)

- Copiez les dossier *tpl* et *images* du répertoire *modules/tgg\_atos/* vers le dossier *themes/votre\_theme/modules/tgg\_atos* (qui doit être créé au besoin) de votre Prestashop.
- Copiez le fichier *modules/tgg\_atos/fr.php* vers *themes/votre\_theme/modules/tgg\_atos/fr.php*
- Editez *themes/votre\_theme/modules/tgg\_atos/fr.php*, remplacez  
`<{tgg_atos}prestashop>`  
au début de chaque ligne par  
`<{tgg_atos}votre_theme>`  
pour conserver les traductions françaises.
- Les fichiers templates du front office sont les fichiers *themes/votre\_theme/modules/tgg\_atos/tpl/tgg\_atos-front-\**
- Les logos des banques sont dans le dossier *themes/votre\_theme/modules/tgg\_atos/images/bank\_logo/*
- Les logos des moyens de paiement sont dans le dossier */card\_logo/*, vous pouvez les remplacer. Si vous les déplacez, n'oubliez pas de paramétrer le chemin vers les logos depuis la page de configuration graphique du module.

Le formulaire contenant les logos des moyens de paiement ne peut être modifié, il est généré par l'exécutable *request*.

**L'emplacement des fichiers de configuration (fs) est trop long (54 caractères maximum).**

Si vous obtenez cette erreur, cela signifie que le dossier en question (par défaut */path\_de\_votre\_prestashop/modules/tgg\_atos/param*) est placé trop bas dans l'arborescence, ou que les noms de certains dossiers parents sont trop longs.

Lorsque vous rencontrez cette erreur, l'appel des exécutable pour certaines banques (voir toutes si vous dépassez de beaucoup les 54 caractères) provoquera une erreur et le module ne fonctionnera donc pas avec ces banques.

Pour résoudre le problème, deux solutions :

- copier ou déplacer le dossier vers un endroit dont le chemin absolu est de moins de 54 caractères, si possible hors de la racine web (améliore la sécurité). Mettre à jour le chemin vers ce dossier depuis le back office du module.
- Créer un lien symbolique vers le dossier depuis un point plus élevé de l'arborescence des fichiers et remplacer le chemin vers le dossier par le chemin absolu du lien symbolique dans le back office module.

## Ajout d'une nouvelle devise au module :

Pour ajouter une nouvelle devise au module il vous faudra éditer le fichier `tgg_atos/tgg_atos.php`, et rechercher le tableau des devises

```
private $_currencies = array(  
    'EUR' => array('978', 2),  
    'USD' => array('840', 2),  
    'CHF' => array('756', 2),  
    'GBP' => array('826', 2),  
    'CAD' => array('124', 2),  
    'JPY' => array('392', 0),  
    'MXN' => array('484', 2),  
    'TRY' => array('949', 2),  
    'AUD' => array('036', 2),  
    'NZD' => array('554', 2),  
    'NOK' => array('578', 2),  
    'BRL' => array('986', 2),  
    'ARS' => array('032', 2),  
    'KHR' => array('116', 2),  
    'TWD' => array('901', 2),  
    'SEK' => array('752', 2),  
    'DKK' => array('208', 2),  
    'KRW' => array('410', 0),  
    'SGD' => array('702', 2),  
    'XPF' => array('953', 0),  
    'XOF' => array('952', 0)  
);
```

Les devises y sont déclarées de la façon suivante :

`'CODE_ISO_ALPHA' => array('CODE_ISO_NUMERIQUE', DECIMALES)`

(avec une virgule à la fin sauf pour la dernière déclaration de devise).

La valeur **DECIMALES** définit le nombre de décimales nécessaires pour afficher la plus petite unité de la devise. Exemple:

Pour l'*Euros*, la plus petite unité est le *cent* (ou *centime* en français), valant 0,01€, donc deux décimales.

Pour le *Yen*, la plus petite unité est le *Yen* lui-même, donc aucune décimale.

Le système Atos/SIPS n'exploite pas le **CODE\_ISO\_ALPHA**, mais celui-ci est nécessaire pour faire le pont avec les devises de Prestashop.

Codes ISO des devises :

[http://www.iso.org/iso/fr/support/faqs/faqs\\_widely\\_used\\_standards/widely\\_used\\_standards\\_other/currency\\_codes/currency\\_codes\\_list-1.htm](http://www.iso.org/iso/fr/support/faqs/faqs_widely_used_standards/widely_used_standards_other/currency_codes/currency_codes_list-1.htm)

## Ajout d'une nouvelle banque au module

Pour ajouter une nouvelle devise au module il vous faudra éditer le fichier *tgg\_atos/tgg\_atos.php*, et rechercher les tableaux de déclaration des banques :

```
private $_banks = array(
    'cyberplus' => 'CyberPlus - Banque Populaire',
    'etransactions' => 'E-Transactions - Crédit Agricole',
    'elysnet' => 'ElysNet - CCF/HSBC',
    'mercanet' => 'Mercanet - BNP',
    'scelliusnet' => 'ScelliusNet - La Banque Postale',
    'sherlocks' => 'Sherlocks - LCL',
    'sogenactif' => 'Sogenactif - Société Générale',
    'webaffaires' => 'WebAffaires - Crédit du Nord'
);
private $_demoCertificates = array(
    'cyberplus' => '038862749811111',
    'etransactions' => '013044876511111',
    'elysnet' => '014102450311111',
    'mercanet' => '082584341411111',
    'scelliusnet' => '014141675911111',
    'sherlocks' => '014295303911111',
    'sogenactif' => '014213245611111',
    'webaffaires' => '014022286611111'
);
```

La clef de chacun des deux tableaux correspond à l'extension du fichier parmcom de la banque et sera utilisée pour prendre le logo de la banque dans le dossier */images/bank\_logo/*.

La valeur du premier tableau est le nom de la banque dans la liste déroulante du back-office.

La valeur du second tableau correspond à l'ID marchand du compte de démonstration de la banque.

Ajoutez donc une ligne par banque à ajouter dans chacun des tableaux.

Ajoutez le logo de banque dans le dossier */images/bank\_logo/* si vous les exploitez dans vos templates.

Ajoutez le fichier parmcom et le fichier certif dans le dossier */param/*.

Vous pouvez également m'envoyer les ressources (logo, fichier parmcom, fichier certif de démonstration) pour que la banque soit intégrée à la version suivante du module.

## ***Passer le module en production***

Lorsque vous êtes satisfait de la configuration de votre module en mode démonstration, vous pouvez commencer le passage en pré-production. Pour cela, envoyez le logo de votre boutique à votre banque et demandez leur le nom de ce logo sur leur serveur pour le saisir dans le champs approprié de la page de configuration graphique du module.

Obtenez ensuite votre fichier de certificat de production (généralement un exécutable contenant le fichier encrypté, exécutez-le, entrez le mot de passe fourni par la banque pour obtenir le fichier).

Basculez le module en mode pré-production, placez votre fichier certificat de production via le formulaire qui apparaît, entrez votre ID marchand si celui-ci vous est demandé, puis sélectionnez-le dans la liste des ID marchands (correspondant aux certificats de production trouvés dans votre dossier */param/*). Si l'utilisation du formulaire de mise en ligne pose un problème quelconque vous pouvez alternativement le mettre en ligne vous même (par FTP, SCP...) dans le dossier.

Une fois l'ID marchand sélectionné, votre module de paiement est en mode production.

Tant que vous n'aurez pas suivi la procédure de passage en production imposé par votre banque, vous serez en mode pré-production: les vérifications d'autorisation de paiement seront réelles mais les sommes ne seront pas débitées. La différence entre production et pré-production se situe uniquement sur le serveur bancaire, le module ne peut influencer sur le basculement entre ces modes et aucune configuration supplémentaire du module n'est nécessaire lors de la bascule entre mode pré-production et production.

## **Questions fréquentes**

### **Ma boutique est en maintenance et la commande n'apparaît pas dans le back-office si je ne clique pas sur le bouton de retour à la boutique, mon module est-il mal installé ?**

Si ce problème n'intervient que lorsque le site est en maintenance, vous avez très probablement simplement omis de saisir les IP des serveurs bancaires dans les IP autorisées à accéder au front-office de votre boutique lors de la maintenance. Sans cela, la banque envoie bien la réponse mais elle ne peut être reçue par le module.

### **Le mode debug indique un code de retour 139 lors de l'appel à l'un des exécutable, qu'est-ce que cela signifie ?**

Là, même Google aurait pu vous répondre ;-)

Il est fort vraisemblable que vous n'avez pas la version des binaires ATOS correspondant à votre système d'exploitation. Mettez votre hébergeur et votre banque en relation pour obtenir les binaires adaptés à votre hébergement.

### **Lorsque j'utilise le mode démonstration, mon numéro de carte bancaire est refusé, mon compte en banque est pourtant loin d'être en négatif, pourquoi ?**

Félicitation pour le solde largement positif de votre compte, compte tenu du manque de donations sur le module je ne peux pas en dire autant ;-)

Plus sérieusement : merci de lire la documentation ATOS que votre banque doit vous fournir lors de la souscription de votre contrat VAD. En mode démonstration on n'utilise pas de vrais numéros de cartes bancaires mais des numéros spécifiques à ce mode. Plus d'informations dans la documentation ATOS, n'étant pas une documentation libre de droit je ne suis pas autorisé à la fournir avec le module. Contactez votre banque s'ils ne vous l'ont pas fournie.

## ***Développer autour de ce module***

Ce module ne met pas d'API à disposition des autres modules et ne stock aucune information en base à part les ID de transaction utilisés sur la journée.

Cependant ce module propose des hooks permettant d'intercepter les données bancaires pour traitement et stockage par les modules ayant besoin de ces informations.

### **Hook `tggAtosBankReturn`**

Ce hook est déclenché après vérification de la cohérence des données envoyées par la banque, quelque soit le vecteur de réponse (automatique/silencieuse ou retour client). Dans le cas où les données sont incohérentes le hook n'est pas déclenché.

Le paramètre du hook est un tableau contenant :

<i>Clé</i>	<i>Valeur</i>
Response	Objet <code>stdClass</code> contenant les valeurs retournées par la banque en tant que propriétés.  Contient en plus des valeurs retournées par la banque: - une propriété <code>'orgine'</code> signalant le type de vecteur de réponse, soit <code>'client_return'</code> , soit <code>'silent_response'</code> . - une propriété <code>'caller_ip_address'</code> contenant l'IP de la machine ayant appelé le script de retour (serveur de banque ou client acheteur).
Cart	Objet <code>Cart</code> représentant le panier dont fait objet le paiement.
Customer	Objet <code>Customer</code> représentant le client acheteur propriétaire du panier.

## Hook `tggAtosOrderConfirm`

Ce hook n'est appelé que lorsque d'un retour bancaire déclenche la validation d'une commande.  
Appelé après validation de la commande.

<i>Clé</i>	<i>Valeur</i>
Response	Objet <code>stdClass</code> contenant les valeurs retournées par la banque en tant que propriétés.  Contient en plus des valeurs retournées par la banque: - une propriété ' <i>orgine</i> ' signalant le type de vecteur de réponse, soit ' <i>client_return</i> ', soit ' <i>silent_response</i> '. - une propriété ' <i>caller_ip_address</i> ' contenant l'IP de la machine ayant appelé le script de retour (serveur de banque ou client acheteur).
Cart	Objet <code>Cart</code> représentant le panier dont fait objet le paiement.
Customer	Objet <code>Customer</code> représentant le client acheteur propriétaire du panier.
Order	Objet <code>Order</code> représentant la commande venant d'être validée.