Informations importantes relatives à votre certificat :

Le certificat commerçant est une clef de sécurité unique permettant à chaque commerçant de communiquer de manière chiffrée avec les serveurs sécurisés d'Atos Worldline.
Aussi, le commerçant est responsable de sa conservation.

Le certificat vous est délivré de manière sécurisée, il vous appartient d'en assurer la conservation, et ainsi :

- D'en restreindre l'accès sur votre serveur
- De le sauvegarder de manière chiffrée
- De ne jamais le copier sur un disque non sécurisé
- De ne jamais l'envoyer (e-mail, courrier) de manière non sécurisée

La compromission d'un certificat et son utilisation par un tiers malveillant perturberait le fonctionnement normal de la boutique, et pourrait notamment :

- générer des transactions non justifiées sur le site du commerçant
- provoquer des opérations de caisse injustifiées (des remboursements par exemple)

Aussi, en cas de compromission de certificat, le commerçant est tenu d'en demander au plus vite la révocation puis le renouvellement à notre service clients.

Pour vous aider à conserver votre certificat de manière sécurisée, vous trouverez ci-après quelques règles à respecter impérativement.

REGLE IMPERATIVE : Protection de vos accès FTP

Toutes les consignes énoncées ci-dessous seront sans intérêt si un pirate a facilement accès au serveur FTP ou aux fichiers du site Web marchand. Il est donc important de le protéger par un mot de passe qui respecte toutes les règles de sécurité, donc qu'il soit suffisamment compliqué pour ne pas être deviné ou retrouvé. Il convient par ailleurs de ne point le divulguer et il est fortement conseillé de le modifier régulièrement.

OPTION 1 : Installation du certificat dans un répertoire non internet

Cette première option est la plus sûre pour résoudre les problèmes de paramétrage de sécurité.
Elle consiste à créer un répertoire non accessible par un navigateur web, donc placé à la racine du serveur (au dessus du répertoire /www/) et d'y installer le certificat.

OPTION 2 : Mesures de protection et de gestion d'accès au fichier

Si la première option est impossible (cas d'hébergements mutualisés par exemple), d'autres solutions de sécurisation du certificat sont possibles :

- **Protection globale bloquant la lecture des répertoires sur site internet (type Apache)**

  Mettre un fichier .htaccess à la racine du site internet contenant une interdiction d'indexation pour toute requête de type POST ou GET.

- **Protection de répertoire par password (type Apache)**

  Vous pouvez demander une authentification par password pour permettre l'accès à certains répertoires. Pour cela il convient de créer 2 types de fichiers : .htaccess et .htpasswd

  Le fichier **.htaccess** est à placer à la racine du répertoire que vous souhaitez protéger. Il contient des directives d'authentification et le chemin d'accès au fichier .htpasswd contenant les couples user/password.

  Le fichier **.htpasswd** doit être installé dans un répertoire non accessible par un navigateur web (au dessus du répertoire /www/).

- **Blocage des robots indexeurs**

  Afin d'éviter l'indexation par les robots des répertoires, il convient de définir un fichier **robots.txt** directement interprétable par les robots indexeurs (Google…).

  Ce fichier permet d'interdire le référencement du contenu d'un répertoire entier en une seule opération. Il suffit donc d'y indiquer le répertoire où est installé le certificat et de lui attribuer une directive « Disallow ».

- **Restriction des droits du fichier**

  Pour les commerçants sous IIS (serveur http de Microsoft), le panneau contrôle web permet de bloquer en lecture/écriture l'accès à tous les répertoires du site internet. Il convient donc de demander à l'hébergeur de configurer les droits d'accès du ou des répertoires à protéger.

  Toutefois, la protection du répertoire peut ne pas être suffisante. Mettre les droits r--r----- sur le fichier certif devrait permettre de ne pas pouvoir afficher son contenu si quelqu'un cherche à y accéder directement.

Informations importantes relatives à la page d'appel à l'API Payment :

La page de votre site qui affiche les logos des moyens de paiement acceptés est la page d'appel à l'API Payment :

Cette page est particulièrement sensible car elle précède directement la redirection de l'internaute vers la page de saisie des coordonnées bancaires.

L'API Payment vous a été livrée avec un fichier d'exemple minimaliste d'appel à l'API (fichier call_request[extension] ou parfois RequestServlet.java). Il vous appartient d'y ajouter des règles de sécurité en cohérence avec le fonctionnement de votre site pour :

- Empêcher que cette page soit affichable sans initialisation préalable de paiement sur votre site
- Protéger le passage des paramètres sensibles que reçoit cette page (comme le montant)

La non sécurisation de cette page peut permettre à un tiers malveillant d'en exploiter les failles pour :

- modifier le montant de son paiement à son avantage
- générer des transactions non justifiées sur le site du commerçant
- envoyer des requêtes en rafale et accéder facilement à la page suivante de saisie des données bancaires

Aussi, en cas de constat d'accès malveillants sur la page d'appel à l'API Payment de son site e-commerce, le commerçant est tenu de corriger au plus vite la faille exploitée.

Pour vous aider à protéger la page d'appel à l'API Payment de votre site, vous trouverez ci-après quelques conseils qui pourront se montrer précieux.

LE PASSAGE DE PARAMETRES

Dans le cas de l'utilisation d'un formulaire pour réaliser le passage de paramètres vers la page d'appel à l'API Payment, la méthode privilégiée pour envoyer la requête devrait être la méthode « POST ». Exemple :

```
<form action= "call_request.php" method="post">
```

De plus, l'utilisation des champs cachés d'un formulaire pour le passage de ces mêmes paramètres est à proscrire car leur valeur est facilement modifiable par quelqu'un de malveillant juste après le clic sur le logo de la carte :

```
<input type="hidden" name="amount " value="100 ">
```

LE CONTROLE D'ACCES SUR LA PAGE D'APPEL A L'API PAYMENT

- Les sessions :

L'idéal pour la sécurisation de cette page serait d'obliger l'internaute à s'inscrire sur le site pour y faire un paiement. De plus, certaines informations permettant d'identifier cet internaute pourraient être placées en session. L'accès à la page d'appel à l'API Payment serait alors conditionné au fait qu'une session ait bien été précédemment initialisée par l'internaute grâce à son authentification.

Si la procédure d'inscription d'un internaute n'est pas possible sur l'e-commerce en question ou ne correspond pas aux besoins du site, il est toujours possible d'utiliser les sessions pour y placer des informations spécifiques à la transaction en cours. Les systèmes de génération de « jetons de reconnaissance » (ou « tokens ») s'avèrent par exemple très efficaces. Le but est tout simplement de s'assurer que la requête provient bien du serveur la demandant, et non d'un autre site éloigné. Là encore, un accès direct d'une personne malveillante à la page d'appel à l'API Payment serait bloqué en constatant qu'il manque des informations en session ou que le jeton n'est pas présent.

- Le referer :

Le contrôle du champ « REFERER » de la requête HTTP arrivant sur la page d'appel à l'API Payment est une autre manière de vérifier que l'internaute qui arrive sur cette page vient bien de notre site et non d'un site extérieur de façon malveillante. Ce système simple fonctionne dans la majorité des cas mais reste toutefois contournable et est moins fiable qu'une protection par « token ».

LES SCRIPTS D'EXEMPLES LIVRES AVEC L'API PAYMENT

L'API Payment est livrée avec plusieurs scripts d'exemples. Parmi eux, il y a le script qui affiche les logos des moyens de paiements acceptés sur le site (celui dont on parle ci-dessus). Les valeurs renseignées dans les exemples de l'API doivent absolument être modifiées lorsque les scripts sont installés sur l'environnement de production du site e-commerce. Elles ne doivent pas non plus faire office de valeurs par défaut en cas de problème quelconque rencontré pendant l'exécution du script. Le script devrait plutôt afficher un message d'erreur et empêcher de poursuivre la cinématique de paiement ou l'exécution de la suite des événements.