TLS 1.0 Information sur le protocole

2017,

Bonjour,

Pour votre information, **Payline désactivera le protocole TLS 1.0 le 4 avr. 2017 en homologation et début 2018 pour les URLs de production**. Cette désactivation est nécessaire pour préserver le haut niveau de sécurité des paiements. Cette article a pour but de vous aider à gérer cette transition pour vos propres serveurs (connexion aux API Payline), mais aussi pour vos acheteurs (pages de paiement et widget) s'ils devaient rencontrer des difficultés.

Qu'est-ce que le TLS ?

TLS signifie « Transport Layer Security » (sécurité de couche de transport). Il s'agit d'un protocole qui fournit la confidentialité et l'intégrité des données entre deux applications qui communiquent. Ce protocole de sécurité est actuellement le plus largement déployé et utilisé. Il est utilisé par les navigateurs Web et les autres applications qui doivent sécuriser l'échange de données sur un réseau. Le TLS garantit qu'une connexion vers un point de terminaison distant est dirigée vers la destination voulue à l'aide du cryptage et de la vérification de l'identité du point de terminaison. Les versions actuelles du TLS sont 1.0, 1.1 et 1.2.
Les page Web et les API de Payline utilisent le TLS en tant que principal dispositif de sécurité.

En quoi consiste cette modification ?

Payline désactivera la version 1.0 de ce protocole aux dates suivantes :

- 4 avr. 2017 en **homologation** pour toutes les URLs (API et pages de paiement)
- 31 mars 2018 en **production** pour les connexions aux API (services.payline.com et services-cc.payline.com)
- 29 Mai 2018 en **production** pour les pages de paiement

A ces dates les clients et vos serveurs ne pourront plus utiliser ce protocole pour accéder à Payline.

Comment les acheteurs seront impactés ?

Certains navigateurs ou mobiles anciens ne supportent pas les versions de TLS supérieur à 1.0. Dans ce cas les acheteurs ne pourront pas afficher les pages de paiement ou le widget. Ils obtiendront des messages d'erreur indiquant qu'une connexion sécurisée avec le serveur Payline n'a pas pu être établie. Les acheteurs devront mettre à jour leur logiciel pour accéder à Payline. Vous trouverez ci-dessous une liste de compatibilité par navigateur.

Comment vos serveurs seront impactés ?

Si vos serveurs ne supportent pas le TLS 1.1 ou 1.2 vous ne pourrez plus accéder aux API Payline et initier des demandes de paiement. Vous devez dans ce cas, soit activer explicitement ces versions sur vos serveurs soit effectuer une session de "patch management" pour mettre à jour les versions des logiciels utilisés, et le rendre compatible avec TLS 1.1 ou 1.2.

A partir du 4 avr. 2017 le TLS 1.0 sera désactivé sur les serveurs d'homologation. Nous vous recommandons vivement d'effectuer un simple appel vers nos API pour valider la compatibilité de vos serveurs. Si vous obtenez des erreurs du type SSL error, handshake failure... C'est très certainement que vos serveurs ne sont pas à jour. Dans ce cas contactez notre support pour qu'il vous aide dans cette mise à jour. Vous devrez appliquer ces modifications avant le mars 2018.

Vous trouverez ci-dessous une liste de compatibilité par environnement technique.

Liste de compatibilité des navigateurs

Cette liste se veut la plus précise possible, mais certains navigateurs et mobile peuvent manquer

Navigateurs/équipements	Compatible TLS 1.1 ou supérieur	Commentaire
IE 11 Desktop et Mobile	Oui	Si le message d'erreur « Une sécurité plus élevée est requise » est affiché, il peut être nécessaire de désactiver le paramètre TLS 1.0 dans la liste Options Internet | Paramètres avancés.
IE 8, 9, 10	Partiel	Compatible uniquement lorsqu'il est exécuté sous Windows 7 ou plus récent, mais pas par défaut. Vous devez activer le TLS 1.1 et 1.2. Sous XP, il n'est pas compatible.

IE 7 et inférieur	Non	
IE 10 et inférieur sur mobile	Non	
Microsoft Edge	Oui	
Firefox 27 et supérieur	Oui	
Firefox 23 à 26	Partiel	Compatible, mais pas par défaut. Utilisez about:config pour activer le TLS 1.1 ou TLS 1.2 en mettant à jour la valeur de configuration security.tls.version.max sur 2 pour le TLS 1.1 ou sur 3 pour le TLS 1.2.
Firefox 22 et inférieur	Non	
Google Chrome 38 et supérieur	Oui	
Google Chrome 22 à 37	Partiel	Compatible lorsqu'il est exécuté sous Windows XP SP3, Vista ou plus récent (pour ordinateur de bureau), OS X 10.6 (Snow Leopard) ou plus récent (pour ordinateur de bureau), ou Android 2.3 (Gingerbread) ou plus récent (pour appareil mobile).
Google Chrome 21 et inférieur	Non	
Navigateur Android 5.0 (Lollipop) et supérieur	Oui	
Navigateur Android 4.4 (KitKat) à 4.4.4	Partiel	Peut être compatible avec le TLS 1.1 ou supérieur. Il se peut que certains appareils équipés d'Android 4.4.x ne prennent pas en charge le TLS 1.1 ou supérieur.
Navigateur Android 4.3 (Jelly Bean) et inférieur	Non	
Safari pour ordinateur de bureau versions 7 et supérieures pour OS X 10.9 (Mavericks) et supérieur	Oui	
Safari pour ordinateur de bureau versions 6 et inférieures pour OS X 10.8 (Mountain Lion) et inférieur	Non	
Safari mobile versions 5 et supérieures pour iOS 5 et supérieur	Oui	
Safari mobile pour iOS 4 et inférieur	Non	

Liste de compatibilité des environnements techniques

Plate-forme ou bibliothèque	**Compatible TLS 1.1 ou supérieur**	**Commentaire**
Java 8 (1.8) et supérieur	Oui	

Java 7 (1.7)	Partiel	Activez les TLS 1.1 et TLS 1.2 à l'aide de la propriété système Java https.protocols pour HttpsURLConnection. Pour activer les TLS 1.1 et TLS 1.2 dans des connexions non-HttpsURLConnection, définissez les protocoles activés dans les instances SSLSocket et SSLEngine créées au sein du code source de l'application.
Java 6 (1.6) mise à jour 111 et supérieure	Partiel	Activez le TLS 1.1 à l'aide de la propriété système Java https.protocols pour HttpsURLConnection. Pour activer le TLS 1.1 dans des connexions non-HttpsURLConnection, définissez les protocoles activés dans les instances SSLSocket et SSLEngine créées au sein du code source de l'application.
Java 6 (1.6) et inférieure	Non	
.NET 4.6 et supérieur	Oui	
.NET 4.5 à 4.5.2	Partiel	.NET 4.5, 4.5.1 et 4.5.2 n'activent pas par défaut les TLS 1.1 et TLS 1.2. Deux options permettent de les activer, comme suit : **Option 1 :** Les applications .NET peuvent activer directement les TLS 1.1 et TLS 1.2 dans leur code logiciel en définissant System.Net.ServicePointManager.SecurityProtocol pour activer SecurityProtocolType.Tls12 et SecurityProtocolType.Tls11. Voici un exemple avec le code C# : System.Net.ServicePointManager.SecurityProtocol = SecurityProtocolType.Tls12 | SecurityProtocolType.Tls11 | SecurityProtocolType.Tls; **Option 2 :** Il peut être possible d'activer par défaut le TLS 1.2 sans modifier le code source, en définissant la valeur DWORD SchUseStrongCrypto sur 1 dans les deux clés de registre suivantes, en les créant si elles n'existent pas : « HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\.NETFramework\v4.0.30319 » et « HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Microsoft\.NETFramework\v4.0.30319 ». Bien que le numéro de version de ces clés de registre soit 4.0.30319, les infrastructures .NET 4.5, 4.5.1 et 4.5.2 utilisent également ces valeurs. Cependant, ces clés de registre activent par défaut le TLS 1.2 dans toutes les applications .NET 4.0, 4.5, 4.5.1 et 4.5.2 de ce système. Par conséquent, nous recommandons de tester cette modification avant de la déployer vers vos serveurs de production. Elle est également disponible en tant que fichier d'importation du registre. Ces valeurs de registre n'affectent pas les applications .NET qui définissent la valeur System.Net.ServicePointManager.SecurityProtocol.
.NET 4.0	Partiel	.NET 4.0 n'active pas par défaut le TLS 1.2. Pour activer le TLS 1.2 par défaut, il est possible d'installer .NET Framework 4.5, ou une version plus récente, et de définir la valeur DWORD SchUseStrongCrypto sur 1 dans les deux clés de registre suivantes, en les créant si elles n'existent pas : « HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\.NETFramework\v4.0.30319 » et « HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Microsoft\.NETFramework\v4.0.30319 ». Cependant, ces clés de registre peuvent activer par défaut le TLS 1.2 dans toutes les applications .NET 4.0, 4.5, 4.5.1 et 4.5.2 de ce système. Nous recommandons de tester cette modification avant de la déployer vers vos serveurs de production. Elle est également disponible en tant que fichier d'importation du registre. Ces valeurs de registre n'affectent pas les applications .NET qui définissent la valeur System.Net.ServicePointManager.SecurityProtocol.
.NET 3.5 et inférieur	Non	

Python 2.7.9 et supérieur	Oui	
Python 2.7.8 et inférieur	Non	
Ruby 2.0.0	Oui	
Ruby 1.9.3 et inférieur	Partiel	Le symbole :TLSv1_2 n'existe pas dans 1.9.3 et inférieur, mais il est possible d'appliquer un correctif à Ruby pour ajouter ce symbole et de compiler Ruby avec OpenSSL 1.0.1 ou supérieur.
Windows Server 2012 R2 et supérieur Windows 8.1 et supérieur	Oui	
Windows Server 2008 R2 à 2012 Windows 7 et 8	Partiel	Compatible par défaut si Internet Explorer 11 est installé. Si Internet Explorer 8, 9 ou 10 est installé, un utilisateur ou un administrateur doit activer les TLS 1.1 et TLS 1.2.

Windows Server 2008 et inférieur Windows Vista et inférieur	Non	
OpenSSL 1.0.1 et supérieur	Oui	
OpenSSL 1.0.0 et inférieur	Non	