E-TRANSACTIONS

APPEL PAR CLE HMAC

VERSION DU 14/09/2015

AVERTISSEMENT

Les informations contenues dans ce document n'ont aucune valeur contractuelle. Elles peuvent faire l'objet de modification à tout moment. Elles sont à jour en date de rédaction au 01/03/2015.

E-transactions est une solution d'encaissement et de gestion des paiements à distance par carte bancaire, dans un environnement sécurisé, distribuée par les Caisses régionales de Crédit Agricole.
Renseignez-vous auprès de votre conseiller sur les conditions générales et tarifaires de cette solution.

ASSISTANCE

Pour tout renseignement ou assistance à l'installation et à l'utilisation de nos produits, nos Equipes restent à disposition des commerçants et Intégrateurs, du lundi au vendredi de 9H à 18H30 :

Support Technique & Fonctionnel :

E-mail : support@e-transactions.fr

Téléphone : 0 810 812 810 *(1)*

(1)prix d'un appel local non surtaxé depuis un poste fixe

Pour tout contact auprès de nos services, il faut impérativement se munir de ses identifiants E-Transactions :

- numéro de SITE (7 chiffres) ;
- numéro de RANG (2 chiffres) ;
- numéro d'identifiant E-Transactions (1 à 9 chiffres).

TABLE DES MATIERES

INTRODUCTION **2**

1. CREATION DE LA CLE **3**

A. GENERATION 3
B. VALIDATION 4
C. EXPIRATION 4
D. TRANSMISSION 4

2. APPEL DE LA PLATEFORME DE PAIEMENT **5**

A. CHOIX DU SERVEUR A APPELER 5
B. CALCUL DE LA SIGNATURE HMAC 5

3. NOUVELLES VARIABLES **7**

A. PBX_HASH 7
B. PBX_HMAC 7
C. PBX_TIME 7

4. EXEMPLES DE CODE **8**

INTRODUCTION

Depuis 2012, E-Transactions a mis en place une nouvelle méthode pour appeler la plateforme de paiement, l'appel HMAC. Contrairement à l'ancienne méthode, il n'est plus nécessaire d'installer un module sur le serveur commerçant.

1. CREATION DE LA CLE

Cette clé est indispensable, elle permet d'authentifier tous les messages échangés entre le site Marchand et les serveurs E-Transactions. Le commerçant doit donc générer sa propre clé unique et confidentielle et l'utiliser pour calculer une empreinte sur ses messages.

a. Génération

L'interface de génération de la clé secrète d'authentification se trouve dans l'onglet « Paramètres » du Back Office Commerçant Vision.

Voici à quoi ressemble cette interface

Le champ « Phrase de passe » peut être renseigné avec une phrase, un mot de passe, ou tout autre texte.

La « Qualité de la phrase » est mise à jour automatiquement lorsque la phrase de passe est saisie. Ces champs permettent de définir des règles d'acceptation minimales de la phrase de passe. Les règles fixées actuellement demandent une phrase de passe d'au moins 15 caractères de long et d'une force de 90%. Le bouton « Générer la clé» restera grisé tant que ces limitations ne sont pas respectées.

La force de la phrase de passe est calculée selon certains critères spécifiques, à savoir le nombre de majuscules, minuscules, caractères spéciaux, etc. Il conviendra donc de varier les caractères saisis, de les alterner et d'éviter les répétitions qui tendent à diminuer le score final.

Le bouton « Générer une clé » permet de calculer la clé d'authentification à partir de la phrase de passe saisie. Ce calcul est une méthode standard assurant le caractère aléatoire de la clé et renforçant sa robustesse. Cette méthode de calcul étant fixe, il est possible à tout moment de retrouver sa clé en retapant la même phrase de passe et en relançant le calcul.

> *Attention, il est possible que le calcul de la clé prenne quelques secondes, selon le navigateur Internet utilisé et la puissance de l'ordinateur. Au cours du calcul, il se peut que le navigateur Internet Explorer demande s'il faut « arrêter l'exécution de ce script ». Il faut répondre « Non » à cette alerte, et patienter jusqu'à la fin du calcul.*

Une fois le calcul terminé, la clé sera affichée dans le champ « Clé ». Il est alors possible de copier/coller cette clé d'authentification pour l'intégrer dans la base de données du site Marchand, ou autre mode de stockage sécurisé.

Il est également possible de saisir dans le champ « Clé » sa propre clé d'authentification (au format

hexadécimal) qui aurait été calculée grâce à un autre moyen que cette interface. La taille minimale de la clé à saisir correspond à une génération de clé en SHA-1, soit 40 caractères hexadécimaux. Cependant, si cette méthode de saisie d'une clé d'authentification « externe » est utilisée, une alerte s'affichera pour rappeler que le Crédit Agricole ne peut pas en garantir la robustesse.

Le bouton « Générer la clé » est grisé par défaut. Les 2 actions qui peuvent activer le bouton sont :

- Saisir une phrase de passe de plus de 15 caractères et dont la force est de plus de 90%
- Saisir une clé hexadécimale de plus de 40 caractères.

Après validation du formulaire, un message récapitulatif sera affiché sur la page, expliquant qu'un email de demande de confirmation a été envoyé à votre adresse mail. La clé qui vient d'être générée ne sera pas active tant que les indications de validation décrites dans cet email n'auront pas été appliquées.
La clé est affichée sur ce récapitulatif. Pour des raisons de sécurité, cette clé ne sera plus transmise ni demandée par nos services. Par conséquent, si cette clé est égarée, il sera nécessaire d'en générer une nouvelle. Il est donc important de veiller à copier la clé d'authentification affichée avant de quitter la page.

> *La clé est dépendante de la plateforme sur laquelle elle est générée. Cela signifie qu'il faut générer une clé pour l'environnement de test et une pour l'environnement de production.*

b. Validation

Une fois l'enregistrement de la nouvelle clé effectué, un email de demande de confirmation vous sera envoyé. Dans cet email se trouvera un lien pointant sur le programme « CBDValid.cgi », par exemple :

https://admin.e-transactions.fr/cgi/CBDValid.cgi?id=5475C869BB64B33F35D0A37DF466568475BC9601

Le paramètre « id » n'est pas la clé saisie, il s'agit d'un « token » généré aléatoirement qui correspond à la clé à valider. Comme dit précédemment, la clé ne sera pas transmise dans l'email.
Après avoir cliqué sur ce lien, si un message annonce « Votre clé est activée », alors la clé est immédiatement en fonction. Ce qui signifie que la clé qui vient d'être validée devrait aussi être en fonction sur le site Marchand.

c. Expiration

Lorsque la clé est validée, celle-ci se voit affectée une date d'expiration (1 an).
Quand cette date sera atteinte, la clé ne sera pas directement désactivée, pour permettre au site Marchand de continuer à fonctionner, mais vous serez averti par email et sur la page d'accueil du back-office E-transactions que cette clé est expirée. Il est fortement recommandé de générer une nouvelle clé d'authentification dans ce cas-là.

d. Transmission

La clé secrète d'authentification ne doit en aucun cas être transmise par e-mail. Le Crédit Agricole ne vous la demandera jamais. Vous devez donc être particulièrement vigilants quant aux demandes suspectes de transmission de la clé d'authentification, il s'agit probablement d'une tentative de phishing ou social engineering.

En cas de perte de la clé secrète, nous ne serons donc pas en mesure de la redonner, il faudra donc en générer une nouvelle.

2. APPEL DE LA PLATEFORME DE PAIEMENT

a. Choix du serveur à appeler

La liste des URL des serveurs E-Transactions est détaillée dans le tableau suivant :

Plate-forme	URL d'accès	Adresses IP publiques affectées aux infrastructures E-transactions	Adresses IP sortantes depuis les infrastructures E-transactions
Pré-production	https://preprod-tpeweb.e-transactions.fr/cgi/**MYchoix_pagepaiement.cgi**	195.25.7.149	195.101.99.76
Production	https://tpeweb.e-transactions.fr/cgi/**MYchoix_pagepaiement.cgi** **Ou** https://tpeweb1.e-transactions.fr/cgi/**MYchoix_pagepaiement.cgi**	194.2.160.69 194.2.160.76 194.2.160.85 194.2.160.92 195.25.7.149 195.25.7.158 195.25.67.5 195.25.67.12	194.2.122.158 194.2.122.190 195.25.7.166 195.25.67.22

En cas d'indisponibilité du serveur principal, il est possible de basculer sur le serveur secondaire. Il est de la responsabilité du site Marchand de vérifier la disponibilité d'une URL avant de rediriger le client, afin d'assurer une disponibilité maximale de son service.

Il est possible de tester la disponibilité des serveurs en essayant d'accéder à une page HTML « load.htm ». Cette page contient uniquement la chaîne « OK » qui confirme que le serveur est accessible.

Un exemple de code PHP permettant de sélectionner un serveur disponible est fourni dans la partie « Exemple de codes ».

b. Calcul de la signature HMAC

Afin de sécuriser le paiement, c'est-à-dire assurer que c'est bien le commerçant qui en est à l'origine et que personne de malveillant n'a modifié une variable (le montant par exemple), E-Transactions propose désormais une authentification par empreinte HMAC.

- Etape 0 : Si ce n'est déjà fait, le commerçant doit générer une clé secrète via l'accès Back-Office commerçant. La procédure est décrite dans le paragraphe précédent.
- Etape 1 : il faut ensuite, lors de la création d'un message à destination des serveurs E-Transactions, concaténer l'ensemble des variables en séparant chaque variable par le symbole « & ». Pour le message ci-dessus (§4.1), il faut donc se baser sur la chaine suivante :

```
PBX_SITE=1999888&PBX_RANG=32&PBX_IDENTIFIANT=2&PBX_TOTAL=1000&PBX_DEVISE=978&PBX_CMD=TEST
&PBX_PORTEUR=test@paybox.com&PBX_RETOUR= Mt:M;Ref:R;Auto:A;Erreur:E
&PBX_HASH=SHA512&PBX_TIME=2011-02-28T11:01:50+01:00
```

- Etape 2 : il est alors possible de lancer le calcul de l'empreinte HMAC en utilisant
    - La chaine qui vient d'être construite
    - La clé secrète obtenue via le Back Office
    - Un algorithme au choix (cf. PBX_HASH)
- Etape 3 : le résultat obtenu (l'empreinte) doit alors être placé dans le champ PBX_HMAC de la requête.

- L'ordre dans la chaine à hasher doit être strictement identique à l'ordre des variables dans le formulaire.
- Dans la chaine à hasher, il faut utiliser les données « brutes », c'est-à-dire ne pas utiliser les fonctions d'encodage URL
- Si vous utilisez déjà l'ancienne méthode de communication avec E-Transactions (par module CGI sur le serveur marchand), le premier appel HMAC bloquera les paiements par l'ancienne méthode.

Une fois la signature HMAC calculée, l'ensemble des variables (dont PBX_HMAC) doit être concaténé et transmis directement au serveur E-Transactions via la méthode HTTP POST.

3. NOUVELLES VARIABLES

a. PBX_HASH

Format : Texte. **Obligatoire**.

Valeur par défaut : SHA512

Définit l'algorithme de hachage utilisé lors du calcul du HMAC.

Cet algorithme doit être choisi parmi la liste suivante :

- SHA512
- SHA224
- SHA384
- RIPEMD160
- SHA256
- MDC2

Les hachages en MD2/4/5 sont jugés trop faibles pour être utilisés, nous ne les accepterons donc pas.

PBX_HASH doit être renseigné avec une des valeurs de cette liste, en respectant la casse (majuscules), et doit bien entendu correspondre au hachage utilisé pour le calcul du HMAC.

Si PBX_HASH n'est pas renseigné mais que dans les trames d'appel la variable PBX_HMAC est quand même renseignée (voir ci-dessous), l'algorithme de hachage sélectionné sera SHA512.

b. PBX_HMAC

Format : Texte (format hexadécimal). **Obligatoire**.

Permet l'authentification du commerçant et la vérification de l'intégrité du message. Il est calculé à partir de la liste des autres variables envoyées à la plateforme.

c. PBX_TIME

Format : Date au format ISO8601. **Obligatoire**.

Date à laquelle l'empreinte HMAC a été calculée. Doit être URL-encodée.

4. EXEMPLES DE CODE

L'extrait de code suivant permet de calculer la clé HMAC et fournit le formulaire permettant d'appeler la plateforme de paiement :

```
<?php
// On récupère la date au format ISO-8601
$dateTime = date("c");
// On crée la chaîne à hacher sans URLencodage
$msg = "PBX_SITE=1999888".
"&PBX_RANG=32".
"&PBX_IDENTIFIANT=2".
"&PBX_TOTAL=".$_POST['montant'].
"&PBX_DEVISE=978".
"&PBX_CMD=".$_POST['ref'].
"&PBX_PORTEUR=".$_POST['email'].
"&PBX_RETOUR=Mt:M;Ref:R;Auto:A;Erreur:E".
"&PBX_HASH=SHA512".
"&PBX_TIME=".$dateTime;


// On récupère la clé secrète HMAC (stockée dans une base de données sécurisée par exemple) et
que l'on renseigne dans la variable $keyTest;

// Si la clé est en ASCII, On la transforme en binaire
$binKey = pack("H*", $keyTest);

// On calcule l'empreinte (à renseigner dans le paramètre PBX_HMAC) grâce à la fonction hash_hmac
et // la clé binaire
// On envoie via la variable PBX_HASH l'algorithme de hachage qui a été utilisé (SHA512 dans ce
cas)
// Pour afficher la liste des algorithmes disponibles sur votre environnement, décommentez la
ligne // suivante
// print_r(hash_algos());

$hmac = strtoupper(hash_hmac('sha512', $msg, $binKey));
// La chaîne sera envoyée en majuscules, d'où l'utilisation de strtoupper()

// On crée le formulaire à envoyer
// ATTENTION : l'ordre des champs est extrêmement important, il doit
// correspondre exactement à l'ordre des champs dans la chaîne hachée
?>
<form method="POST" action="https://urlserveur.paybox.com/cgi/MYchoix_pagepaiement.cgi">
<input type="hidden" name="PBX_SITE" value="1999888">
<input type="hidden" name="PBX_RANG" value="32">
<input type="hidden" name="PBX_IDENTIFIANT" value="2">
<input type="hidden" name="PBX_TOTAL" value="<? echo $_POST['montant']; ?>">
<input type="hidden" name="PBX_DEVISE" value="978">
<input type="hidden" name="PBX_CMD" value="<? echo $_POST['ref']; ?>">
<input type="hidden" name="PBX_PORTEUR" value="<? echo $_POST['email']; ?>">
<input type="hidden" name="PBX_RETOUR" value="Mt:M;Ref:R;Auto:A;Erreur:E">
<input type="hidden" name="PBX_HASH" value="SHA512">
<input type="hidden" name="PBX_TIME" value="<? echo $dateTime; ?>">
<input type="hidden" name="PBX_HMAC" value="<? echo $hmac; ?>">
<input type="submit" value="Envoyer">
</form>
```

L'extrait de code suivant permet de vérifier la disponibilité des serveurs afin d'appeler un serveur disponible :

```
<?php
	$serveurs = array('tpeweb.paybox.com', //serveur primaire
			'tpeweb1.paybox.com'); //serveur secondaire

	$serveurOK = "";
	phpinfo();
	foreach($serveurs as $serveur){
		$doc = new DOMDocument();
		$doc->loadHTMLFile('https://'.$serveur.'/load.html');

		$server_status = "";
		$element = $doc->getElementById('server_status');
		if($element){
			$server_status = $element->textContent;
		}

		if($server_status == "OK"){
			//Le serveur est prêt et les services opérationnels
			$serveurOK = $serveur;
			break;
		}
		// else : La machine est disponible mais les services ne le sont pas.
	}
	curl_close($ch);


	if(!$serveurOK){
		die("Erreur : Aucun serveur n'a été trouvé");
```