Réussir l'intégration de l'API formulaire

Guide d'implémentation

Version du document 1.2

Sommaire

1. CONTACTER L'ASSISTANCE TECHNIQUE

Pour toute question technique ou demande d'assistance, nos services sont disponibles de 07h30 à 17h30

par téléphone au :	(687) 46 33 33
par e-mail :	*sav@csb.nc*

Pour faciliter le traitement de vos demandes, il vous sera demandé de communiquer votre identifiant de boutique (numéro à 8 chiffres) .

Cette information est disponible dans l'e-mail d'inscription de votre boutique ou dans le Back Office (menu **Paramétrage** > **Boutique** > **Configuration**).

2. ÉTABLIR LE DIALOGUE AVEC LA PLATEFORME DE PAIEMENT

Le dialogue entre le site marchand et la plateforme de paiement s'effectue par un échange de données.

Pour créer un paiement, ces données sont envoyées au moyen d'un formulaire HTML via le navigateur de l'acheteur.

A la fin du paiement, le résultat est transmis au site marchand de deux manières :

- automatiquement au moyen de notifications appelées URL de notification instantanée (également appelée IPN pour Instant Payment Notification) voir chapitre **Configurer la notification à la fin du paiement**.
- par le navigateur lorsque l'acheteur clique sur le bouton pour revenir au site marchand.

Pour assurer la sécurité des échanges, les données sont signées au moyen d'une clé (anciennement appelée "certificat") connue uniquement du marchand et de la plateforme de paiement.

2.1. Définir l'URL de la page de paiement

Le site marchand communique avec la plateforme de paiement en redirigeant l'acheteur vers la page :

https://epaync.nc/vads-payment/

2.2. S'identifier lors des échanges

Pour dialoguer avec la plateforme de paiement, le marchand a besoin de deux informations :

- **L'identifiant boutique** : permet d'identifier le site marchand durant les échanges. Sa valeur est transmise dans le champ **vads_site_id.**
- **La clé**: anciennement appelée "certificat", permet de calculer la signature alphanumérique transmise dans le champ **signature**.

Pour récupérer ces valeurs :

1. Connectez-vous à votre Back Office : *https://epaync.nc/vads-merchant/*
2. Cliquez sur **Paramétrage > Boutique**.
3. Sélectionnez l'onglet **Certificats**.

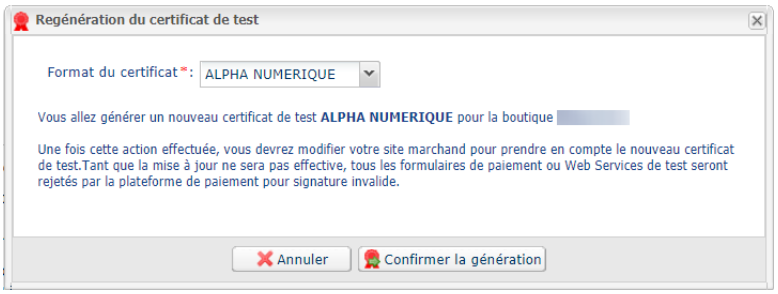
Image 1 : Onglet Certificats

Deux types de clé (certificat) sont mis à disposition :

- La **clé (certificat) de test** qui permet de générer la signature d'un formulaire en mode test.
- La **clé (certificat) de production** qui permet de générer la signature d'un formulaire en mode production.

Ces clés peuvent être numériques ou alphanumériques.

Pour changer le format de votre clé de test, cliquez sur le bouton **Régénérer un certificat de test**, puis sélectionnez le format ("ALPHANUMERIQUE" ou "NUMERIQUE").



Pour changer le format de votre clé de production, cliquez sur le bouton **Régénérer un certificat de production**, puis sélectionnez "ALPHANUMERIQUE" ou "NUMERIQUE").

2.3. Gérer le dialogue vers le site marchand

La gestion du dialogue vers le site marchand est réalisée grâce à deux types d'URL :

- **Url de notification instantanée**, également appelée IPN (Instant Payment Notification),
- **Url de retour** vers le site marchand.

Url de notification instantanée - IPN (Instant Payment Notification)

La plateforme de paiement notifie automatiquement au site marchand le résultat du paiement. Les données sont envoyées en mode **POST**.

La plateforme est capable de contacter le site marchand quel que soit le protocole utilisé (http ou https).

Url de retour vers le site marchand

Le marchand peut paramétrer dans le Back Office les URL de retour "par défaut" depuis le menu **Paramétrage** > **Boutique** > onglet **Configuration** :

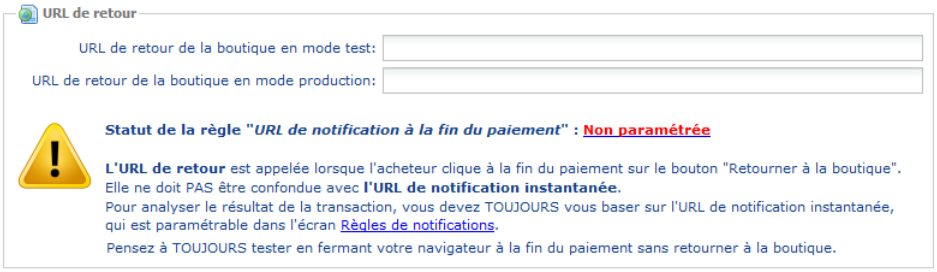


Image 2 : Spécification des URL de retour

Il peut configurer une URL de retour à la boutique différente en fonction du mode.

Par défaut, l'acheteur est redirigé vers l'URL de retour, et ce, quel que soit le résultat du paiement.

Si toutefois aucune URL n'est configurée à ce niveau, alors la redirection utilisera l'URL principale de la boutique (paramètre **URL** défini dans l'encadré **Détails** de la boutique).

Remarque :

Le statut de la règle "URL de notification à la fin du paiement" (IPN) est affiché dans cet écran. Si cette dernière est non paramétrée, veillez à la renseigner (voir chapitre **Paramétrer les notifications**) .

2.4. Gérer la sécurité

Plusieurs moyens sont mis en place afin d'assurer la sécurité des transactions de paiement en ligne.

Garantir l'intégrité des échanges

L'intégrité des informations échangées est garantie par un échange de signatures alphanumériques entre la plateforme de paiement et le site marchand.

Le dialogue entre la plateforme de paiement et le site marchand s'effectue par soumission de formulaires HTML.

Un formulaire contient une liste de champs spécifiques (voir chapitre **Générer un formulaire de paiement**) utilisés pour générer une chaîne.

Cette chaîne est ensuite convertie en une chaîne d'une taille inférieure grâce à la fonction de hachage SHA-256.

La chaîne résultante est appelée **empreinte** (*digest* en anglais) de la chaîne initiale.

L'empreinte doit être transmise dans le champ **signature** (voir chapitre **Calculer la signature**).

Modélisation des mécanismes de sécurité :

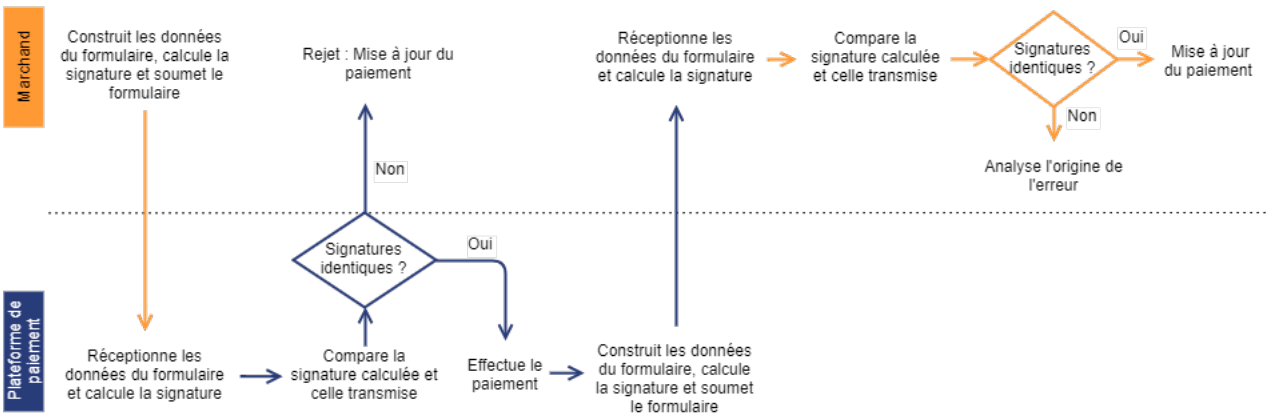


Image 3 : Diagramme mécanisme de sécurité

1. Le site marchand récolte les données du formulaire et calcule la signature.
2. Le site marchand soumet le formulaire à la plateforme.
3. La plateforme réceptionne les données du formulaire et calcule la signature.
4. La plateforme compare la signature calculée avec la signature transmise par le site marchand.
5. Si les signatures diffèrent, la demande de paiement est rejetée.

   Sinon, la plateforme procède au paiement.
6. La plateforme réceptionne les données du résultat et calcule la signature de la réponse.
7. En fonction du paramétrage de la boutique (voir chapitre **Paramétrer les notifications**), la plateforme soumet le résultat du paiement au site marchand.

8. Le site marchand réceptionne les données et calcule la signature. Il compare la signature calculée avec la signature transmise par la plateforme.
9. Si les signatures diffèrent, le marchand analyse l'origine de l'erreur (erreur dans le calcul, tentative de fraude etc.)

   Sinon, le site marchand procède à la mise à jour de sa base de données (état du stock, état de la commande etc.).

Conserver la clé de production

Dès le premier paiement réalisé avec une carte réelle, la clé (certificat) de production est masqué pour des raisons de sécurité.

Nous vous conseillons fortement de conserver cette clé en lieu sûr (fichier chiffré, base de données etc.).

En cas de perte, le marchand aura la possibilité d'en générer une nouvelle depuis son Back Office.

Pour rappel, la clé de production est visible dans le Back Office depuis le menu **Paramétrage** > **Boutique** > onglet **Certificats**.

Gérer les données sensibles

Des règles strictes régissent les transactions de paiement en ligne (Certification PCI-DSS).

En tant que marchand, vous devez vous assurer de ne jamais retranscrire en clair des données qui pourraient s'apparenter à un numéro de carte bancaire. Votre formulaire serait rejeté (code 999 - Sensitive data detected).

Evitez notamment les numéros de commandes de longueur comprise entre 13 et 16 caractères numériques et commençant par 3, 4 ou 5.

2.5. Configurer la notification à la fin du paiement

Cette notification est indispensable pour communiquer le résultat d'une demande de paiement.

Pour paramétrer cette notification :

1. Effectuez un clic droit sur la ligne **URL de notification à la fin du paiement.**
2. Sélectionnez **Gérer la règle**.
3. Renseignez l'URL de votre page dans les champs **URL à appeler en mode TEST** et **URL à appeler en mode PRODUCTION.**

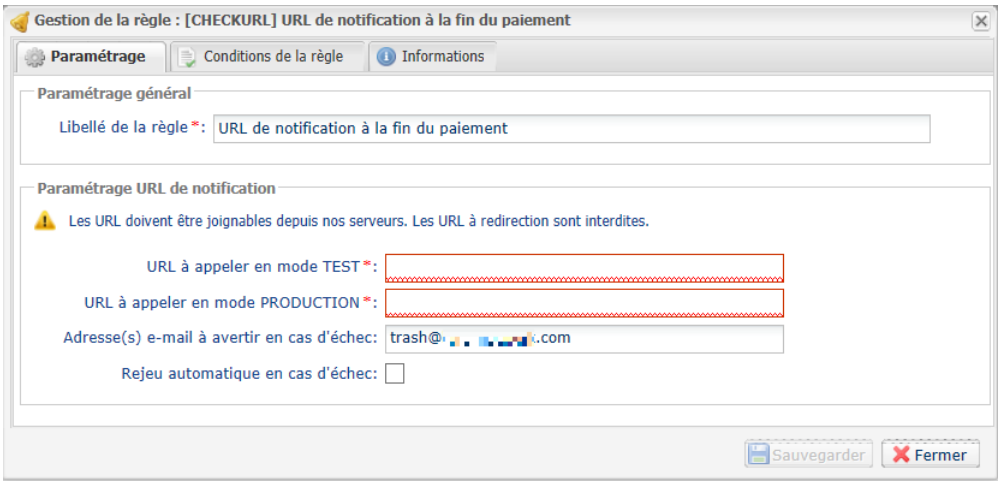


Image 4 : URL de notification à la fin du paiement

4. Renseignez le champ **Adresse(s) e-mail(s) à avertir en cas d'échec**.
5. Pour spécifier plusieurs adresses e-mails, séparez-les par un point-virgule.
6. Configurez le **Rejeu automatique en cas d'échec**.

   Cette option permet de renvoyer automatiquement la notification vers le site marchand en cas d'échec, et ce, jusqu'à 4 fois.

   Pour plus d'informations, reportez-vous au chapitre **Activer le rejeu automatique** du guide d'implémentation du formulaire disponible sur notre site documentaire
7. Sauvegardez vos modifications.

Si la plateforme n'arrive pas à joindre l'URL de votre page, alors un e-mail est envoyé à l'adresse spécifiée à l'étape 4.

Il contient :

- Le code HTTP de l'erreur rencontrée
- Des éléments d'analyse en fonction de l'erreur
- Ses conséquences
- La procédure à suivre depuis le Back Office pour renvoyer la requête vers l'URL définie à l'étape 6.

Autres cas de notification

En fonction des options commerciales souscrites, la plateforme de paiement pourra effectuer un appel vers l'url de notification dans les cas suivants :

- abandon ou annulation de la part de l'acheteur sur la page de paiement
- remboursement effectué depuis le Back Office
- annulation d'une transaction depuis le Back Office
- validation d'une transaction depuis le Back Office
- modification d'une transaction depuis le Back Office

Pour plus d'informations, reportez-vous au chapitre **Paramétrer les notifications** du guide d'implémentation du formulaire disponible sur notre site documentaire :

https://epaync.nc/espace-commercant/.

3. ENVOYER UN FORMULAIRE DE PAIEMENT EN POST

Le site marchand redirige l'acheteur vers la plateforme de paiement sous la forme d'un formulaire HTML POST en HTTPS.

Ce formulaire contient :

Les éléments techniques suivants :

- Les balises `<form>` et `</form>` qui permettent de créer un formulaire HTML.
- L'attribut `method="POST"` qui spécifie la méthode utilisée pour envoyer les données.
- L'attribut `action="https://epaync.nc/vads-payment/"` qui spécifie où envoyer les données du formulaire.

Les données du formulaire :

Toutes les données du formulaire doivent être encodées en **UTF-8.**

Les caractères spéciaux (accents, ponctuation etc.) seront ainsi correctement interprétés par la plateforme de paiement. Dans le cas contraire, le calcul de signature sera erroné et le formulaire sera rejeté.

- Les champs obligatoires :

Nom du champ	Description	Valeur
vads_site_id	Identifiant de la boutique	Ex : 12345678
vads_ctx_mode	Mode de fonctionnement	**TEST** ou **PRODUCTION**
vads_trans_id	Numéro de la transaction	Ex : 123456
vads_trans_date	Date et heure du formulaire de paiement dans le fuseau horaire UTC	Ex : 20170129130025
vads_amount	Montant du paiement	Ex : 3000 pour 3000 XPF
vads_currency	Code numérique de la monnaie à utiliser pour le paiement, selon la norme ISO 4217 (code numérique).	Ex : 953 pour le Franc CFP (XPF)
vads_action_mode	Mode d'acquisition des données de la carte	**INTERACTIVE**
vads_page_action	Action à réaliser	**PAYMENT**
vads_version	Version du protocole d'échange	**V2**
vads_payment_config	Type de paiement	**SINGLE** pour un paiement en 1 fois **MULTI** pour un paiement en plusieurs fois
signature	Signature unique pour chaque paiement	Ex: vSlCWjJwN8TpobRyuyKhwAlKEhlThtICZiI/ rmpPK4U=

Tableau 1 : Liste des champs obligatoires

- Les champs recommandés :
  - Les données de la commande

Nom du champ	Description	Format	Valeur
vads_order_id	Numéro de commande	ans..64	Ex : `2-XQ001`
vads_order_info	Informations supplémentaire sur la commande	an..255	
vads_order_info2	Informations supplémentaire sur la commande	an..255	
vads_order_info3	Informations supplémentaire sur la commande	an..255	
vads_nb_products	Nombre d'articles présents dans le panier	n..12	Ex : `2`

Nom du champ	Description	Format	Valeur

vads_product_labelN	Libellé de l'article. N correspond à l'indice de l'article (0 pour le premier, 1 pour le second...)	an..255	Ex :
			`vads_product_label0 = "tee-shirt"`
			`vads_product_label1 = "Biscuit"`
			`vads_product_label2 = "sandwich"`

vads_product_amountN	Montant de l'article. N correspond à l'indice de l'article (0 pour le premier, 1 pour le second...)	n..12	Ex :
			`vads_product_amount0 = "1200"`
			`vads_product_amount1 = "800"`
			`vads_product_amount2 = "950"`

vads_product_typeN	Type de l'article. N correspond à l'indice de l'article (0 pour le premier, 1 pour le second...)	enum	Ex :
			`vads_product_type0 = "CLOTHING_AND_ACCESSORIES"`
			`vads_product_type1 = "FOOD_AND_GROCERY"`
			`vads_product_type2 = "FOOD_AND_GROCERY"`

vads_product_refN	Référence de l'article. N correspond à l'indice de l'article (0 pour le premier, 1 pour le second...)	an..64	Ex :
			`vads_product_ref0 = "CAA-25-006"`
			`vads_product_ref1 = "FAG-B5-112"`
			`vads_product_ref2 = "FAG-S9-650"`

vads_product_qtyN	Quantité d'article. N correspond à l'indice de l'article (0 pour le premier, 1 pour le second...)	n..12	Ex :
			`vads_product_qty0 = "1"`
			`vads_product_qty1 = "2"`
			`vads_product_qty2 = "2"`

Tableau 2 : Liste des champs - Détails de la commande

- Les données de l'acheteur

Nom du champ	Description
vads_cust_email	Adresse e-mail de l'acheteur.
vads_cust_id	Référence de l'acheteur sur le site marchand.
vads_cust_title	Civilité de l'acheteur.
vads_cust_status	Statut (**PRIVATE**: pour particulier / **COMPANY** pour une entreprise).
vads_cust_first_name	Prénom.
vads_cust_last_name	Nom.
vads_cust_legal_name	Raison sociale de l'acheteur.
vads_cust_cell_phone	Numéro de téléphone mobile.
vads_cust_phone	Numéro de téléphone.
vads_cust_address_number	Numéro de rue.
vads_cust_address	Adresse postale.
vads_cust_district	Quartier.
vads_cust_zip	Code postal.
vads_cust_city	Ville.
vads_cust_state	Etat / Région.
vads_cust_country	Code pays suivant la norme ISO 3166.

Tableau 3 : Liste des champs - Détails de l'acheteur

- Les données de livraison

Nom du champ	Description	Format	Valeur
vads_ship_to_city	Ville	an..128	Ex : Papeete
vads_ship_to_country	Code pays suivant la norme ISO 3166	a2	Ex : PF

Nom du champ	Description	Format	Valeur
vads_ship_to_district	Quartier	ans..127	Ex : Mission
vads_ship_to_first_name	Prénom	ans..63	Ex : Moana
vads_ship_to_last_name	Nom	ans..63	Ex : Doom
vads_ship_to_legal_name	Raison sociale	an..100	Ex : D. & Cie
vads_ship_to_name	**Déprécié**. Nom de l'acheteur. Utilisez **vads_ship_to_first_name** et **vads_ship_to_last_name**.	ans..63	
vads_ship_to_phone_num	Numéro de téléphone	ans..32	Ex: 40975711
vads_ship_to_state	Etat / Région	ans..127	Ex : Tahiti
vads_ship_to_status	Définit le type d'adresse de livraison	enum	**PRIVATE**: pour une livraison chez un particulier **COMPANY** pour une livraison en entreprise
vads_ship_to_street_number	Numéro de rue	ans..64	Ex : 2
vads_ship_to_street	Adresse postale	ans..255	Ex : Impasse Cardela
vads_ship_to_street2	Deuxième ligne d'adresse	ans..255	
vads_ship_to_zip	Code postal	an..64	Ex : `98713`

Tableau 4 : Liste des champs - Détails de la livraison

- Les champs facultatifs :

Vous pouvez utiliser des paramètres facultatifs supplémentaires.

Référez-vous au chapitre **Dictionnaire de données** du guide d'implémentation du formulaire disponible sur notre site documentaire (*https://epaync.nc/espace-commercant/*) afin de visusaliser la liste des champs disponibles.

Vous pourrez notamment :

- Décrire le contenu du panier et de la commande
- Envoyer des informations sur l'acheteur et sur la livraison
- Personnaliser les pages de paiement

Le bouton **Payer** qui va permettre l'envoi des données :

```
<input type="submit" name="payer" value="Payer"/>
```


4. CALCULER LA SIGNATURE

Afin de pouvoir calculer la signature vous devez être en possession :

- de la totalité des champs dont le nom commence par **vads**_
- du type d'algorithme choisi dans la configuration de la boutique
- de la **clé (certificat)**

La valeur de la clé est disponible dans votre Back Office depuis le menu **Paramétrage** > **Boutique** > onglet **Certificats**.

Le type d'algorithme est défini dans votre Back Office depuis le menu **Paramétrage** > **Boutique** > onglet **Configuration**.

Pour calculer la signature :

1. Triez les champs dont le nom commence par **vads**_ par ordre alphabétique.
2. Assurez-vous que tous les champs soient encodés en UTF-8.
3. Concaténez les valeurs de ces champs en les séparant avec le caractère "**+**".
4. Concaténez le résultat avec la clé de test ou de production en les séparant avec le caractère "**+**".
5. Appliquez l'algorithme SHA-256 sur la chaîne obtenue en utilisant la clé (de test ou de production en fonction de la valeur du champ vads_ctx_mode) comme clé partagée.
6. Encodez le résultat en base64.

Exemple de paramètres envoyés à la plateforme de paiement:

```
<form method="POST" action="https://epaync.nc/vads-payment/">
<input type="hidden" name="vads_action_mode" value="INTERACTIVE" />
<input type="hidden" name="vads_amount" value="5124" />
<input type="hidden" name="vads_ctx_mode" value="TEST" />
<input type="hidden" name="vads_currency" value="953" />
<input type="hidden" name="vads_page_action" value="PAYMENT" />
<input type="hidden" name="vads_payment_config" value="SINGLE" />
<input type="hidden" name="vads_site_id" value="12345678" />
<input type="hidden" name="vads_trans_date" value="20170129130025" />
<input type="hidden" name="vads_trans_id" value="123456" />
<input type="hidden" name="vads_version" value="V2" />
<input type="hidden" name="signature" value="vSlCWjJwN8TpobRyuyKhwAlKEhlThtICZiI/rmpPK4U= " />

<input type="submit" name="payer" value="Payer"/>
</form>
```

Cet exemple de formulaire s'analyse de la manière suivante:

1. On trie par ordre **alphabétique** les champs dont le nom commence par **vads**_ :
   - vads_action_mode
   - vads_amount
   - vads_ctx_mode
   - vads_currency
   - vads_page_action
   - vads_payment_config
   - vads_site_id
   - vads_trans_date
   - vads_trans_id
   - vads_version

2. On concatène la valeur de ces champs avec le caractère "**+**" :

```
INTERACTIVE+5124+TEST+953+PAYMENT+SINGLE+12345678+20170129130025+123456+V2
```

3. On ajoute la valeur de la clé de test à la fin de la chaîne en la séparant par le caractère "**+**". Dans cet exemple, la clé de test est **1122334455667788**

```
INTERACTIVE+5124+TEST+953+PAYMENT+SINGLE+12345678+20170129130025+123456+V2+1122334455667788
```

4. Appliquez la fonction de hachage SHA-256 sur la chaîne obtenue en utilisant la clé comme clé partagée, puis encodez le résultat en base64.

   Le résultat à transmettre dans le champ signature est :

   vSlCWjJwN8TpobRyuyKhwAlKEhlThtlCZil/rmpPK4U=

4.1. Exemple d'implémentation en JAVA

Définition d'une classe utilitaire Sha utilisant l'algorithme SHA-256 pour calculer la signature:

```
import javax.crypto.Mac;
import javax.crypto.spec.SecretKeySpec;
import java.io.UnsupportedEncodingException;
import java.security.InvalidKeyException;
import java.security.NoSuchAlgorithmException;
import java.util.Base64;
import java.util.TreeMap;

public class VadsSignatureExample {
    /**
    * Build signature (HMAC SHA-256 version) from provided parameters and secret key.
    * Parameters are provided as a TreeMap (with sorted keys).
    */
    public static String buildSignature(TreeMap<String, String> formParameters, String
        secretKey)
        throws NoSuchAlgorithmException, InvalidKeyException, UnsupportedEncodingException {
      // Build message from parameters
      String message = String.join("+", formParameters.values());
      message += "+" + secretKey;
      // Sign
      return hmacSha256Base64(message, secretKey);
    }

    /**
    * Actual signing operation.
    */
    public static String hmacSha256Base64(String message, String secretKey) throws
            NoSuchAlgorithmException, InvalidKeyException, UnsupportedEncodingException {
        // Prepare hmac sha256 cipher algorithm with provided secretKey
        Mac hmacSha256;
        try {
            hmacSha256 = Mac.getInstance("HmacSHA256");
        } catch (NoSuchAlgorithmException nsae) {
            hmacSha256 = Mac.getInstance("HMAC-SHA-256");
        }
        SecretKeySpec secretKeySpec = new SecretKeySpec(secretKey.getBytes("UTF-8"),
 "HmacSHA256");
        hmacSha256.init(secretKeySpec);
        // Build and return signature
        return
 Base64.getEncoder().encodeToString(hmacSha256.doFinal(message.getBytes("UTF-8")));
    }
}
```

4.2. Exemple d'implémentation en PHP

Exemple de calcul de signature utilisant l'algorithme SHA-256:

```
function getSignature ($params,$key)
{
    /**
     * Fonction qui calcule la signature.
     * $params : tableau contenant les champs à envoyer dans le formulaire.
     * $key : clé de TEST ou de PRODUCTION
     */
    //Initialisation de la variable qui contiendra la chaine à chiffrer
    $contenu_signature = "";

    //Tri des champs par ordre alphabétique
    ksort($params);
    foreach($params as $nom=>$valeur){

        //Récupération des champs vads_
        if (substr($nom,0,5)=='vads_'){

            //Concaténation avec le séparateur "+"
            $contenu_signature .= $valeur."+";
         }
    }
    //Ajout de la clé en fin de chaine
    $contenu_signature .= $key;

    //Encodage base64 de la chaine chiffrée avec l'algorithme SHA-256
    $signature = base64_encode(hash_hmac('sha256',$contenu_signature, $key, true));
    return $signature;
 }
```

5. ANALYSER LE RÉSULTAT DU PAIEMENT

L'URL de notification instantanée (IPN - Instant Payment Notification) permet à la plateforme de paiement de notifier automatiquement au site marchand le résultat du paiement.

Les données sont envoyées en mode POST quel que soit le protocole utilisé (http ou https).

5.1. Récupérer les données retournées dans la réponse

Les données retournées dans la réponse dépendent des paramètres envoyés dans le formulaire de paiement, du type de paiement réalisé et des options de votre boutique. Ces données constituent une liste de champs. Chaque champ contient une valeur réponse. La liste de champs peut être amenée à évoluer.

5.2. Calculer la signature

La signature se calcule selon la même logique utilisée lors de la création du formulaire de paiement.

Tous les champs reçus doivent être pris en compte.

Pour calculer la signature:

1. Prenez en considération la totalité des champs dont le nom commence par **vads**_.
2. Triez ces champs par ordre alphabétique.
3. Assurez-vous que tous les champs soient encodés en UTF-8.
4. Concaténez les valeurs de ces champs en les séparant avec le caractère "**+**".
5. Concaténez le résultat avec la clé de test ou de production en les séparant avec le caractère "**+**".
6. Appliquez l'algorithme SHA-256 sur la chaîne obtenue en utilisant la clé (de test ou de production en fonction de la valeur du champ vads_ctx_mode) comme clé partagée.
7. Encodez le résultat en base64.

5.3. Comparer les signatures

Pour s'assurer de l'intégrité de la réponse, vous devez comparer la valeur du champ **signature** reçue dans la réponse, avec celle calculée à l'étape précédente.

Si les signatures correspondent,

- alors vous pouvez considérer la réponse comme sûre et procéder à la suite de l'analyse.
- sinon, le script devra lever une exception et avertir le marchand de l'anomalie (voir chapitre **Traiter les erreurs** du guide d'implémentation du formulaire disponible sur notre site documentaire :

  https://epaync.nc/espace-commercant/).

Les signatures ne correspondent pas en cas :

- d'erreur d'implémentation (erreur dans votre calcul, problème d'encodage UTF-8, etc.),
- d'erreur dans la valeur de la clé utilisée ou dans celle du champ **vads_ctx_mode** (problème fréquent lors du passage en production),
- de tentative de corruption des données.

5.4. Traiter les données de la réponse

Ci-dessous un exemple d'analyse pour vous guider pas à pas lors du traitement des données de la réponse.

1. Identifiez la commande en récupérant la valeur du champ **vads_order_id** si vous l'avez transmis dans le formulaire de paiement.

   Vérifiez que le statut de la commande n'a pas déja été mis à jour.

2. Récupérez le résultat du paiement transmis dans le champ **vads_trans_status**.

   Sa valeur vous permet de définir le statut de la commande.

Valeur	Description
ABANDONED	**Abandonné** Paiement abandonné par l'acheteur. La transaction n'est pas créée et **n'est donc pas visible dans le Back Office.**
AUTHORISED	**En attente de remise** La transaction est acceptée et sera remise en banque automatiquement à la date prévue.
AUTHORISED_TO_VALIDATE	**A valider** La transaction, créée en validation manuelle, est autorisée. Le marchand doit valider manuellement la transaction afin qu'elle soit remise en banque. La transaction peut être validée tant que la date d'expiration de la demande d'autorisation n'est pas dépassée. Si cette date est dépassée alors le paiement prend le statut **EXPIRED**. Le statut **Expiré** est définitif.
CANCELLED	**Annulée** La transaction est annulée par le marchand.
CAPTURED	**Remisée** La transaction est remise en banque.
CAPTURE_FAILED	La remise de la transaction a échoué. Contactez le Support.
EXPIRED	**Expirée** La date d'expiration de la demande d'autorisation est atteinte et le marchand n'a pas validé la transaction. Le porteur ne sera donc pas débité.
INITIAL	**En attente** Ce statut est spécifique à tous les moyens de paiement nécessitant une intégration par formulaire de paiement en redirection. Ce statut est retourné lorsque : • aucune réponse n'est renvoyée par l'acquéreur ou • le délai de réponse de la part de l'acquéreur est supérieur à la durée de session du paiement sur la plateforme de paiement. Ce statut est temporaire. Le statut définitif sera affiché dans le Back Office aussitôt la synchronisation réalisée.
NOT_CREATED	**Transaction non créée** La transaction n'est pas créée et n'est pas visible dans le Back Office.
REFUSED	**Refusée** La transaction est refusée.
UNDER_VERIFICATION	**Vérification en cours** Spécifique à PayPal

Valeur	Description
WAITING_AUTHORISATION	**En attente d'autorisation** Le délai de remise en banque est supérieur à la durée de validité de l'autorisation.
WAITING_AUTHORISATION_TO_VALIDATE	**A valider et autoriser** Le délai de remise en banque est supérieur à la durée de validité de l'autorisation. Une autorisation 100 XPF a été acceptée. Le marchand doit valider manuellement la transaction afin que la demande d'autorisation et la remise aient lieu.

Tableau 5 : Valeurs associées au champ vads_trans_status

3. Récupérez la référence du paiement transmise dans le champ **vads_trans_id**.

4. Analysez le champ **vads_payment_config** pour déterminer s'il s'agit d'un **paiement comptant** (unitaire) ou d'un **paiement en plusieurs fois**.

   Ce champ peut être valorisé à :

Nom du champ	Valeur pour un paiement comptant	Valeur pour un paiement en plusieurs fois
vads_payment_config	SINGLE	MULTI (dont la syntaxe exacte est MULTI:first=X;count=Y;period=Z)

Tableau 6 : Analyse du champ vads_payment_config

   S'il s'agit d'un paiement en plusieurs fois, identifiez le numéro de l'échéance en récupérant la valeur du champ **vads_sequence_number**.

Valeur	Description
1	Première échéance
2	Deuxième échéance
3	Troisième échéance
n	N échéance

Tableau 7 : Analyse du champ vads_sequence_number

   Remarque :

   *Pour un paiement comptant (unitaire) le champ **vads_sequence_number** est valorisé à **1**.*

5. Récupérez la valeur du champ **vads_trans_date** pour identifier la date du paiement.

6. Récupérez la valeur du champ **vads_capture_delay** pour identifier le nombre de jours avant la remise en banque.

   Ceci vous permettra d'identifier s'il s'agit d'un paiement immédiat ou différé.

7. Récupérez le montant et la devise utilisée. Pour cela, récupérez les valeurs des champs suivants:

Nom du champ	Description
vads_amount	Montant du paiement dans sa plus petite unité monétaire.
vads_currency	Code de la devise utilisée pour le paiement.
vads_change_rate	Taux de change utilisé pour calculer le montant réél du paiement (voir vads_effective_amount).
vads_effective_amount	Montant du paiement dans la devise réellement utilisée pour effectuer la remise en banque.
vads_effective_currenc	Devise dans laquelle la remise en banque va être effectuée.

Tableau 8 : Analyse du montant et de la devise utilisée

8. Récupérez la valeur du champ **vads_auth_result** pour connaître le résultat de la demande d'autorisation.

La liste des codes renvoyés est disponible dans le guide d'implémentation du formulaire de paiement.

Pour vous aider à comprendre le motif du refus, voici une liste des codes fréquemment retournés :

Valeur	Description
03	**Accepteur invalide** Ce code est émis par l'acquéreur. Il correspond à un problème de configuration sur les serveurs d'autorisation. (ex: contrat clos, mauvais code MCC déclaré, etc..). **Pour connaître la raison précise du refus, le marchand doit contacter sa banque.**
05	**Ne pas honorer** Ce code est émis par la banque émettrice de la carte. Il est utilisé dans les cas suivants : • Date d'expiration invalide, • CVV invalide, • crédit dépassé, • solde insuffisant (etc.) **Pour connaître la raison précise du refus, l'acheteur doit contacter sa banque.**
51	**Provision insuffisante ou crédit dépassé** Ce code est émis par la banque émettrice de la carte. Il peut être obtenu si l'acheteur ne dispose pas d'un solde suffisant pour réaliser son achat. **Pour connaître la raison précise du refus, l'acheteur doit contacter sa banque.**
56	**Carte absente du fichier** Ce code est émis par la banque émettrice de la carte. Le numéro de carte saisi est erroné ou le couple numéro de carte + date d'expiration n'existe pas.
57	**Transaction non permise à ce porteur** Ce code est émis par la banque émettrice de la carte. Il est utilisé dans les cas suivants : • l'acheteur tente d'effectuer un paiement sur internet avec une carte de retrait, • le plafond d'autorisation de la carte est dépassé. **Pour connaître la raison précise du refus, l'acheteur doit contacter sa banque.**
59	**Suspicion de fraude** Ce code est émis par la banque émettrice de la carte. Il peut être envoyé suite à une saisie répétée de CVV ou de date d'expiration erronée. **Pour connaître la raison précise du refus, l'acheteur doit contacter sa banque.**
60	**L'accepteur de carte doit contacter l'acquéreur** Ce code est émis par l'acquéreur. Il correspond à un problème de configuration sur les serveurs d'autorisation. Il est utilisé lorsque le contrat commerçant ne correspond pas au canal de vente utilisé. (ex : une transaction e-commerce avec un contrat VAD-saisie manuelle). **Contactez le service client pour régulariser la situation.**

Tableau 9 : Valeurs associées au champ vads_auth_result

9. Récupérez le résultat de l'authentification 3D Secure. Pour cela:

   a. Récupérez la valeur du champ **vads_threeds_enrolled** pour déterminer le statut de l'enrôlement de la carte.

Valeur	Description
Vide	Processus 3DS non réalisé (3DS désactivé dans la demande, marchand non enrôlé ou moyen de paiement non éligible au 3DS).
Y	Authentification disponible, porteur enrôlé.
N	Porteur non enrôlé.
U	Impossible d'identifier le porteur ou carte non éligible aux tentatives d'authentification (ex. Cartes commerciales ou prépayées).

Tableau 10 : Valeurs du champ vads_threeds_enrolled

   b. Récupérez le résultat de l'authentification 3D Secure en récupérant la valeur du champ **vads_threeds_status**.

Valeur	Description
Vide	Authentification 3DS non réalisée (3DS désactivé dans la demande, porteur non enrôlé ou moyen de paiement non éligible au 3DS).
Y	Porteur authentifié avec succès.
N	Erreur d'authentification du porteur.
U	Authentification impossible.
A	Tentative d'authentification mais authentification non réalisée.

Tableau 11 : Valeurs du champ vads_threeds_status

10. Récupérez le résultat des contrôles associés à la fraude en identifiant la valeur du champ **vads_risk_control**. Ce champ est envoyé uniquement si le marchand a:

    - souscrit à l'option « **Aide à la décision** »
    - activé au moins un contrôle depuis son Back Office (menu **Paramétrage** > **Contrôle des risques**).

    Il prend comme valeur une liste de valeurs séparées par un « ; » dont la syntaxe est: **vads_risk_control = control1=result1;control2=result2**

    Les valeurs possibles pour **control** sont :

Valeur	Description
CARD_FRAUD	Contrôle la présence du numéro de carte de l'acheteur dans la liste grise de cartes.
SUSPECT_COUNTRY	Contrôle la présence du pays émetteur de la carte de l'acheteur dans la liste des pays interdits.
IP_FRAUD	Contrôle la présence de l'adresse IP de l'acheteur dans la liste grise d'IP.
CREDIT_LIMIT	Contrôle la fréquence et les montants d'achat d'un même numéro de carte, ou le montant maximum d'une commande.
BIN_FRAUD	Contrôle la présence du code BIN de la carte dans la liste grise des codes BIN.
ECB	Contrôle si la carte de l'acheteur est de type e-carte bleue.
COMMERCIAL_CARD	Contrôle si la carte de l'acheteur est une carte commerciale.
SYSTEMATIC_AUTO	Contrôle si la carte de l'acheteur est une carte à autorisation systématique.
INCONSISTENT_COUNTRIE	Contrôle si le pays de l'adresse IP, le pays émétteur de la carte de paiement, et le pays de l'adresse de l'acheteur sont cohérents entre eux.
NON_WARRANTY_PAYMEN	Contrôle le transfert de responsabilité de la transaction.
SUSPECT_IP_COUNTRY	Contrôle la présence du pays de l'acheteur, identifié par son adresse IP, dans la liste des pays interdits.

Tableau 12 : Liste des contrôles associés à la fraude

Les valeurs possibles pour **result** sont :

Valeur	Description
OK	OK.
WARNING	Contrôle informatif échoué.
ERROR	Contrôle bloquant échoué.

Tableau 13 : Liste des contrôles associés à la fraude

11. Récupérez le type de carte utilisé pour le paiement.

    Deux cas de figures peuvent se présenter:

    - Pour un paiement réalisé avec **une seule carte**. Les champs à traiter sont les suivants:

Nom du champ	Description
vads_card_brand	Marque de la carte utilisée pour le paiement. ex : CB, VISA, VISA_ELECTRON, MASTERCARD, MAESTRO, VPAY
vads_card_number	Numéro de la carte utilisée pour réaliser le paiement.
vads_expiry_month	Mois d'expiration entre 1 et 12 (ex: 3 pour mars, 10 pour octobre).
vads_expiry_year	Année d'expiration sur 4 chiffres (ex : 2023).
vads_bank_code	Code de la banque émettrice

Nom du champ	Description
vads_bank_product	Code produit de la carte
vads_card_country	Code Pays du pays d'émission de la carte (Code alpha ISO 3166-2 ex : France=FR).

Tableau 14 : Analyse de la carte utilisée pour le paiement

- Pour un **paiement fractionné** (c'est-à-dire une transaction utilisant plusieurs moyens de paiement), les champs à traiter sont les suivants :

Nom du champ	Valeur	Description
vads_card_brand	MULTI	Plusieurs types de cartes sont utilisés pour le paiement.
vads_payment_seq	Au format json, voir détails ci-dessous.	Détails des transactions réalisées.

Le champ **vads_payment_seq** (format json) décrit la séquence de paiement fractionné. Il contient les éléments :

1. "trans_id" : identifiant de la transaction global à la séquence de paiement.
2. "transaction" : tableau des transactions de la séquence. Les éléments qui le composent sont les suivants :

Nom du paramètre	Description
amount	Montant de la séquence de paiement.
operation_type	Opération de débit.
auth_number	Numéro d'autorisation. Exemple : 949478
auth_result	Code retour de la demande d'autorisation.
capture_delay	Délai avant remise (en jours). • Pour un paiement par carte bancaire, la valeur de ce paramètre tient compte du délai en nombre de jours avant la remise en banque. Si ce paramètre n'est pas transmis dans le formulaire de paiement, la valeur par défaut définie dans le Back Office sera utilisée.
card_brand	Moyen de paiement utilisé.
card_number	Numéro du moyen de paiement. • Pour un paiement par carte bancaire, le numéro est masqué.
expiry_month	Mois d'expiration du moyen de paiement.
expiry_year	Année d'expiration du moyen de paiement.
payment_certificate	Certificat de paiement.
contract_used	Contrat utilisé pour le paiement.
identifier	Identifiant unique (token/alias) associé à un moyen de paiement.
identifier_status	Présent uniquement si l'action demandée correspond à la création ou à la mise à jour d'un alias. Valeurs possibles:

Valeur	Description
CREATED	La demande d'autorisation a été acceptée. L'alias ou RUM est créé avec succès.
NOT_CREATED	La demande d'autorisation a été refusée. L'alias ou RUM n'est pas créé et n'apparaîtra pas dans le Back Office.
UPDATED	L'alias ou RUM est mis à jour avec succès.
NOT_UPDATED	L'alias ou RUM n'a pas été mis à jour.
ABANDONED	Action abandonnée par l'acheteur (débiteur). L'alias ou RUM n'est pas créé et n'apparaîtra pas dans le Back Office.

presentation_date	Pour un paiement par carte bancaire, ce paramètre correspond à la date de remise en banque souhaitée (au format ISO 8601).
trans_id	Numéro de transaction.

<table>
<tr><th>Nom du paramètre</th><th>Description</th></tr>
<tr><td>ext_trans_id</td><td>Paramètre absent pour le paiement par carte bancaire.</td></tr>
<tr><td>trans_uuid</td><td>Référence unique générée par la plateforme de paiement suite à la création d'une transaction de paiement.
Offre une garantie d'unicité pour chaque transaction</td></tr>
<tr><td>extra_result</td><td>Code numérique du résultat des contrôles de risques.
<table>
<tr><th>Code</th><th>Description</th></tr>
<tr><td>Vide</td><td>Pas de contrôle effectué.</td></tr>
<tr><td>00</td><td>Tous les contrôles se sont déroulés avec succès.</td></tr>
<tr><td>02</td><td>La carte a dépassé l’encours autorisé.</td></tr>
<tr><td>03</td><td>La carte appartient à la liste grise du marchand.</td></tr>
<tr><td>04</td><td>Le pays d’émission de la carte appartient à la liste grise du marchand.</td></tr>
<tr><td>05</td><td>L’adresse IP appartient à la liste grise du marchand.</td></tr>
<tr><td>06</td><td>Le code bin appartient à la liste grise du marchand.</td></tr>
<tr><td>07</td><td>Détection d’une e-carte bleue.</td></tr>
<tr><td>08</td><td>Détection d’une carte commerciale nationale.</td></tr>
<tr><td>09</td><td>Détection d’une carte commerciale étrangère.</td></tr>
<tr><td>14</td><td>Détection d’une carte à autorisation systématique.</td></tr>
<tr><td>20</td><td>Contrôle de cohérence : aucun pays ne correspond (pays IP, pays carte, pays de l’acheteur).</td></tr>
<tr><td>30</td><td>Le pays de l’adresse IP appartient à la liste grise.</td></tr>
<tr><td>99</td><td>Problème technique rencontré par le serveur lors du traitement d’un des contrôles locaux.</td></tr>
</table>
</td></tr>
<tr><td>sequence_number</td><td>Numéro de séquence.</td></tr>
<tr><td>trans_status</td><td>Statut de la transaction.</td></tr>
</table>

Tableau 15 : Contenu de l'objet JSON

Remarque : les transactions annulées sont également présentes dans le tableau.

12. Récupérez toutes les informations concernant le détail de la commande, le détail de l'acheteur et le détail de la livraison.

    Ces données sont présentes dans la réponse que si elles ont été envoyées dans le formulaire de paiement.

    Leur valeur est identique à celle soumise dans le formulaire.

13. Procédez à la mise à jour de la commande.

6. PROCÉDER À LA PHASE DE TEST

Préalablement au passage en production de la boutique, il est nécessaire de réaliser des tests pour s'assurer du bon fonctionnement entre le site marchand et la plateforme de paiement.

Ces tests doivent impérativement être réalisés avant de demander le passage en production.

6.1. Réaliser des tests de paiement

Les demandes de paiement de test adressées via le formulaire HTTP POST doivent:

- Contenir la donnée **vads_ctx_mode** valorisée à **TEST**.
- Utiliser **le certificat de test** précédemment récupéré pour le calcul de la signature.

Différents cas de paiements peuvent être simulés en utilisant les numéros de carte de test précisés sur la page de paiement.

Toutes les transactions réalisées en mode test sont consultables par les personnes habilitées à utiliser le Back Office à l'adresse suivante :

https://epaync.nc/vads-merchant/

Ces transactions sont consultables depuis le menu **Gestion >Transaction de test** situé en haut à gauche du Back Office.

6.2. Tester l'URL de notification instantanée (IPN)

Vérifiez tout d'abord l'état de l'URL de notification instantanée (également appelée IPN) dans le Back Office.

Pour cela:

1. Effectuez un clic droit sur une transaction.
2. Sélectionnez **Afficher le détail de la transaction**.
3. Vérifiez le statut de l'URL de notification instantanée (IPN).
   - Dans le cas où le statut est **Envoyé**, cela signifie que vous avez correctement renseigné l'URL dans le Back Office.
   - Dans le cas où le statut apparaît en **URL non définie**, cela signifie que vous n'avez pas renseigné l'URL dans le Back Office.
     1. Vérifiez l'adresse de l'URL de notification instantanée saisie en mode TEST et PRODUCTION.
     2. Cliquez sur **Paramétrage** > **Règles de notification**.
     3. Renseignez l'URL de notification de paiement instantanée (URL de notification à la fin du paiement).

        Ne saisissez pas une adresse en "localhost". L'appel à cette l'URL se fait de serveur à serveur.
     4. Cliquez sur **Sauvegarder**.
   - Dans le cas où le statut est **Echoué**, se reporter au chapitre **Traiter les erreurs** du guide d'implémentation du formulaire disponible sur notre site documentaire .

7. ACTIVER LA BOUTIQUE EN MODE PRODUCTION

Ce chapitre vous détaille de quelle manière vous pouvez :

- Générer la clé de production.
- Basculer votre site marchand en production.
- Réaliser un premier paiement en production.
- Régénérer le certificat de production (en cas de problème).

7.1. Générer la clé de production

Vous pouvez générer la clé de production depuis le menu **Paramétrage** > **Boutique** > Onglet **Certificats** > bouton **Générer le certificat de production**.

Une fois la clé de production générée, sa valeur apparaît sous l'onglet **Certificats**.

Un e-mail est envoyé à l'interlocuteur en charge du dossier (responsable administratif de la société) pour lui confirmer la génération de la clé de production.

7.2. Basculer le site marchand en production

1. Valorisez le champ **vads_ctx_mode** à **PRODUCTION**.
2. Modifiez la valeur de la clé de test avec la valeur de votre clé de production pour calculer la signature.

   Vous trouverez cette valeur depuis le menu **Paramétrage** > **Boutique** > Onglet **Certificats**.
3. Renseignez correctement l'URL de notification à la fin du paiement en mode PRODUCTION depuis le menu **Paramétrage** > **Règles de notification**.

7.3. Réaliser un premier paiement de production

Nous vous conseillons de vérifier les deux points suivants :

- Le bon fonctionnement en environnement de production de bout-en-bout.

  Pour ce faire, effectuez une transaction réelle.
- Le bon fonctionnement de l'URL de notification de paiement (Url de notification à la fin du paiement) renseignée dans le Back Office.

  Pour ce faire, ne cliquez pas sur le bouton **Retour à la boutique** après un paiement.

  Affichez le détail de la transaction dans le Back Office et vérifiez que le statut de l'URL de notification (Statut URL de notification) est bien **Envoyé.**