

Guide fonctionnel des API Contrat-Privé

Version 1.0 du 28 octobre 2016

1	Introduction	2
1.1	Aperçu des API	2
1.2	Plateformes	2
1.3	Accès aux API	2
1.4	Limitation d'utilisations	3
1.5	Marque, CGU	3
1.6	Assistance technique	3
2	API disponibles	4
2.1	Domaine Connexion	4
2.2	Domaine Gestion des contrats	4
2.3	Domaine Signature des contrats	5
3	Exemples d'utilisation des API	6
3.1	Exemple 1 – injection automatique des contrats dans Contrat-Privé	6
3.2	Exemple 2 – intégration de la signature dans votre extranet	6

1 Introduction

1.1 Aperçu des API

Contrat-Privé est une plateforme de signature en ligne et sur smartphones de contrats et documents de toutes natures.

Le service est accessible :

- Sur le site internet www.contrat-privé.com
- Via des interfaces de programmation (API), qui permettent
 - o d'automatiser des actions en masse (exemple : création de contrats)
 - o d'insérer dans votre propre site internet les fonctions de signature et d'horodatage électronique de documents

Ce document présente les principales fonctionnalités disponibles en mode API, et quelques exemples d'utilisation.

L'utilisation des API est détaillée dans notre documentation technique en ligne, disponible sur www.contrat-privé.com/fr/product/api

1.2 Plateformes

Les API sont disponibles sur :

- sandbox.contrat-privé.com, la plateforme « bac à sable », permettant à tous nos clients de développer et tester les fonctionnalités mises à disposition
- www.contrat-privé.com, la plateforme de production, pour les clients ayant souscrit à l'un des packs de jetons Contrat-Privé éligibles (Pack 1K ou Pack 10K).

1.3 Accès aux API

Pour accéder aux API, vous devez disposer de 2 clés :

- apiKey : clé privée d'accès aux API
- apiUser : clé publique du compte déposant (appelé par la suite utilisateur)

Ces deux clés sont disponibles dans votre compte de déposant.

1.4 Limitation d'utilisations

Les limitations suivantes sont mises en place par défaut. Si vous souhaitez disposer de seuils plus élevés, merci de prendre contact avec les administrateurs Contrat-Privé.

Plateforme Bac à sable :

- pas d'envoi réel de SMS – code valide pour les tests 0000
- signature et horodatage factices des documents (filigrane test ajouté)
- pour tester le paiement :
 - o vous pouvez utiliser ce numéro de CB pour payer les acomptes : 4706750000000009 avec n'importe quelle date d'expiration (dans le futur) et n'importe quel cryptogramme visuel (3 chiffres). Aucun compte ne sera vraiment débité.
 - o vous pouvez utiliser n'importe quel RIB (il y en a un en exemple dans le message d'erreur de format). Aucun compte ne sera vraiment crédité.
- En revanche, les notifications par e-mail sont bien réalisées. Attention à n'utiliser que les adresses e-mail des testeurs, pour éviter que vos clients ne reçoivent ces messages.

Plateforme Production :

- Accès aux API limité à 1.000 requêtes par jour
- Seuils d'échec mis en place :
 - o 5 demandes de renvoi de SMS, par signataire et par contrat
 - o 5 échecs au maximum de signature d'un contrat par un signataire

1.5 Marque, CGU

Il est à noter que la marque « Contrat-Privé » apparaît dans le SMS reçu par le signataire pour obtenir son code de signature et dans la dernière page des documents signés. Pour éviter toute surprise, nous vous recommandons d'informer vos utilisateurs que votre service utilise Contrat-Privé.

Nous vous invitons également à mettre en conformité vos conditions d'utilisation en utilisant tout ou partie de nos CGU (<https://www.contrat-privé.com/fr/product/terms-of-use>).

Pour une utilisation en marque blanche complète au travers d'un projet sur mesure, veuillez nous contacter via : <https://www.contrat-privé.com/fr/product/contact>.

1.6 Assistance technique

Une assistance technique est à votre disposition via le formulaire de contact ou par e-mail à l'adresse

support.api@contrat-privé.com

2 API disponibles

2.1 Domaine Connexion

➤ **checkConnexionRequest**

Permet de tester la connexion avec la plateforme (bac à sable ou production), et la validité des clés `apiKey` et `apiUser` de l'utilisateur sur cette plateforme.

A noter qu'un couple de clés n'est valide que sur une seule des deux plateformes.

2.2 Domaine Gestion des contrats

➤ **contractCreateRequest**

Permet de générer un contrat en fournissant l'ensemble des informations requises :

- la description du contrat (libellé, langue, date limite de signature)
- les documents du contrat (à signer et en annexe)
- l'identité de chacun des signataires, et pour chacun d'entre eux, le montant de l'éventuel acompte à payer
- `notify.signatory` : active ou non les notifications pour les signataires (mode « furtif »)
- `notify.owner` : active ou non les notifications pour le déposant
- `notify.submitMessage` (facultatif) : personnalisation des emails de notification éventuels (ne fonctionne qu'avec le mode « submit », pas le mode « save »)
- deux modes d'actions :
 - Si `action.submit` est vrai : procède à l'enregistrement du contrat et à sa publication immédiate pour signature
 - Sinon: procède à l'enregistrement du contrat comme un brouillon, qu'il faudra valider sur le site www.contrat-prive.com pour l'expédier aux signataires

➤ **contractListRequest**

Permet de connaître le statut de l'ensemble des contrats déposés par l'utilisateur ou d'un sous-ensemble (filtre sur statut, plage de dates).

➤ **contractDetailRequest**

Permet de récupérer l'ensemble des informations du contrat signé, dont

- le statut du contrat et des différents signataires
- `mainDocument.data` : les documents déposés
- `mainDocument.data.sign` : les documents signés et horodatés électroniquement
- `log` : la piste d'audit

➤ **contractCancelRequest**

Permet d'annuler un contrat soumis précédemment et non encore signé par l'ensemble des signataires prévus.

2.3 Domaine Signature des contrats

➤ **signatorySendSmsRequest**

Génère et envoie un SMS à un signataire identifié sur un contrat, et ne l'ayant pas encore signé.

➤ **signatorySendAllSmsRequest**

Génère et envoie un SMS à l'ensemble des signataires du contrat qui ne l'ont pas encore signé.

➤ **signatorySignRequest**

Permet de demander la signature du contrat par l'un des signataires, en fournissant le code reçu par SMS.

Précision : il n'est pas possible de déclencher le paiement d'un acompte par API. Le signataire devra impérativement utiliser le site www.contrat-prive.com pour effectuer son éventuel règlement.

3 Exemples d'utilisation des API

3.1 Exemple 1 – injection automatique des contrats dans Contrat-Privé

Cet exemple illustre un cas où les contrats sont générés à partir de votre propre système d'information (SIRH, CRM, ERP), et sont automatiquement déposés dans Contrat-Privé pour permettre aux destinataires de ces contrats d'y accéder pour les signer en ligne.

Les contrats sont signés dans Contrat-Privé.

Votre application collecte les contrats signés et récupère les documents finaux, pour archivage.

- 1 : appel de **contractCreateRequest** pour créer les contrats, avec publication (*action.submit=true*) notification (*notify.signatory=true* et *notify.owner=true* et, éventuellement un message de notification personnalisé (*notifyMessage*)
- 2 : appel une fois par jour (ou fréquence aux choix) de **contractListRequest**, pour connaître les contrats signés et non encore récupérés
- 3 : appel à **contractDetailRequest** pour récupérer le contrat signé
- 4 : mise à jour de l'application du client

Une variante consiste à préparer des envois de contrat en masse en utilisant l'action *save* au lieu de *submit*. Cela permet une vérification unitaire des contrats dans l'interface Contrat-Privé et une complétion manuelle des informations manquantes dans votre système d'information (le numéro de portable des signataires par exemple) avant envoi.

3.2 Exemple 2 – intégration de la signature dans votre extranet

Cet exemple illustre une intégration totale, en marque blanche, de la fonction de signature (sans acompte) dans votre propre extranet client. Dans cet exemple, les utilisateurs effectuent l'ensemble de leurs actions sur votre site internet, sans nécessiter un passage par www.contrat-privé.com.

- 1 : votre site présente à ses utilisateurs les documents à signer et un bouton Signer Mon Contrat
- 2 : appel de **contractCreateRequest** pour créer le contrat en mode furtif (*action.submit=true* et *notify.signatory=false* et éventuellement *notify.owner=false* aussi)
- 3 : appel de **signatorySendSmsRequest** pour envoyer au signataire son code de signature par SMS
- 4 : l'utilisateur saisit le code sur votre site
- 5 : appel de **signatorySignRequest**, pour signer le contrat avec le code saisi par l'utilisateur
- 6 : appel à **contractDetailRequest** pour récupérer le contrat signé
- 7 : confirmation sur votre site à l'utilisateur que la signature s'est bien passée, et mise à jour de votre système d'information