

ⓘ Cette page a été traduite à partir de l'anglais par la communauté. [Vous pouvez contribuer en rejoignant la communauté francophone sur MDN Web Docs.](#)

[View in English](#)

Always switch to English

API Push

**Baseline** Large disponibilité

ⓘ **Note :** Cette fonctionnalité est disponible via les [Web Workers](#).

L'**API Push** permet aux applications web de recevoir des messages poussés depuis un serveur, que l'application soit active au premier plan, chargée, ou non. Ceci permet de fournir des notifications asynchrones et des alertes/mises à jour pour les personnes qui souhaitent en recevoir, créant l'opportunité d'être au courant rapidement.

Concepts et usages de Push

⚠ **Attention :** Lorsqu'on implémente des abonnements via `PushManager`, il est primordial de se protéger contre les attaques CSRF/XSRF. Pour plus d'informations, voir les articles suivants :

- [Anti-sèche pour se protéger contre les attaques CSRF \(Cross-Site Request Forgery\) \(en anglais\)](#) ↗
- [Prévenir les attaques CSRF et XSRF \(en anglais\)](#) ↗

Pour qu'une application web puisse recevoir des messages *push*, elle doit disposer d'un [service worker](#) actif. Quand le *service worker* est actif, on peut souscrire aux notifications *push* en utilisant `PushManager.subscribe()`.

Cet abonnement fournira un objet `PushSubscription` [\(angl.\)](#) contenant toutes les informations dont l'application a besoin pour transmettre un message *push* : un point de terminaison et la clé de chiffrement nécessaire à l'envoi des données.

Le *service worker* sera alors démarré pour réceptionner les messages *push* via le gestionnaire d'évènement `onpush` [\(angl.\)](#). Ainsi, l'application peut réagir aux messages *push* reçus, par exemple en affichant une notification avec `ServiceWorkerRegistration.showNotification()`.

Chaque abonnement est propre à un *service worker*. Le point d'accès correspondant pour l'abonnement est une [URL fonctionnelle \(capability URL\)](#) ↗ unique. Il suffit de connaître ce point d'accès pour envoyer un message à votre application. L'URL du point d'accès doit donc rester secrète, afin d'éviter que d'autres applications puissent envoyer des messages *push* vers votre application.

L'activation d'un *service worker* nécessaire à la réception du message *push* peut accroître l'utilisation des ressources, particulièrement de la batterie. Chaque navigateur adopte une stratégie différente pour gérer cette consommation, sans qu'il y ait actuellement de mécanisme standard. Firefox impose un quota de messages *push* pouvant être envoyés à une application (cependant, les messages *push* utilisés pour les notifications sont exempts de cette limite). Le quota est réinitialisé à chaque visite sur le site. Chrome n'impose pas de limite particulière.

Interfaces

PushEvent

Représente une action *push* envoyée à la portée globale (angl.) d'un `ServiceWorker`. Elle contient les informations envoyées depuis le serveur d'application vers l'objet `PushSubscription` (angl.) responsable du traitement.

PushManager

Permet de recevoir des notifications de serveurs tiers et de demander les URL pour les notifications *push*.

PushMessageData (angl.)

Fournit l'accès aux données transmises par le serveur, incluant des méthodes pour manipuler les données reçues.

PushSubscription (angl.)

Fournit l'URL de l'abonnement au point d'accès et permet également le désabonnement.

PushSubscriptionOptions (angl.)

Représente les options associées à l'abonnement.

Ajouts à l'API *Service Worker*

Les ajouts à l'API *Service Worker* suivants ont été spécifiés dans l'API *Push*, pour fournir un point d'entrée à l'utilisation de messages `Push` pour surveiller et répondre aux événements de changement liés aux messages `Push` et aux souscriptions.

ServiceWorkerRegistration.pushManager (angl.) Lecture seule

Fournit une référence à l'interface une référence de l'interface `PushManager` permettant de gérer les abonnements *push* (créer un nouvel abonnement, obtenir l'abonnement existant, connaître l'état des permissions, se désabonner). Il s'agit du point d'entrée vers l'API *Push*.

onpush (angl.)

Un gestionnaire d'événement déclenché à chaque événement `push` (angl.) ; c'est-à-dire à chaque message *push* reçu depuis le serveur.

onpushsubscriptionchange (angl.)

Un gestionnaire d'événement déclenché à chaque événement `pushsubscriptionchange` (angl.) ; par exemple, quand un abonnement *push* a été invalidé, ou va l'être (par exemple, quand un service *push* définit une durée d'expiration).

Exemples

Le livre de recettes pour les [service workers](#) contient différents exemples d'utilisation de l'API *Push*.

Spécifications

Spécification
Push API # pushevent-interface
Push API # pushmessagedata-interface

Compatibilité des navigateurs

api.PushEvent

	Chrome	Edge	Firefox	Opera	Safari	Chrome Android	Firefox for Android	Opera Android	Safari on iOS	Samsung Browser	WebView Android	WebView on iOS
PushEvent	✓ 42	✓ 17	✓ 44	✓ 37	✓ 16 *	✓ 42	✓ 48	✓ 37	✓ 16.4 *	✓ 4	✗ Non	✗ Non *
PushEvent() constructor	✓ 42	✓ 17	✓ 44	✓ 37	✓ 16 *	✓ 42	✓ 48	✓ 37	✓ 16.4 *	✓ 4	✗ Non	✗ Non *
data	✓ 50	✓ 17	✓ 44	✓ 37	✓ 16 *	✓ 50	✓ 48	✓ 37	✓ 16.4 *	✓ 5	✗ Non	✗ Non *
notification ⚠	✗ Non *	✗ Non	✗ Non *	✗ Non *	✓ 18.4	✗ Non *	✗ Non *	✗ Non *	✓ 18.4	✗ Non *	✗ Non *	✗ Non *

✓
Prise en charge complète
✗
Pas de prise en charge

⚠
Expérimental. Attendez-vous à ce que les comportements changent à l'avenir.

*
Voir les notes de mise en application.

api.PushMessageData

	Chrome	Edge	Firefox	Opera	Safari	Chrome Android	Firefox for Android	Opera Android	Safari on iOS	Samsung Browser	WebView Android	WebView on iOS
PushMessageData	✓ 50	✓ 17	✓ 44	✓ 37	✓ 16 *	✓ 50	✓ 48	✓ 37	✓ 16.4 *	✓ 5	✗ Non	✗ Non *

arrayBuffer	✓ 50	✓ 17	✓ 44	✓ 37	✓ 16 *	✓ 50	✓ 48	✓ 37	✓ 16.4 *	✓ 5	⊗ Non	⊗ Nor *
blob	✓ 50	✓ 17	✓ 44	✓ 37	✓ 16 *	✓ 50	✓ 48	✓ 37	✓ 16.4 *	✓ 5	⊗ Non	⊗ Nor *
bytes	✓ 132	✓ 132	✓ 128	✓ 117	✓ 18	✓ 132	✓ 128	✓ 87	✓ 18	✓ 29	⊗ Non	⊗ Nor
json	✓ 50	✓ 17	✓ 44	✓ 37	✓ 16 *	✓ 50	✓ 48	✓ 37	✓ 16.4 *	✓ 5	⊗ Non	⊗ Nor *
text	✓ 50	✓ 17	✓ 44	✓ 37	✓ 16 *	✓ 50	✓ 48	✓ 37	✓ 16.4 *	✓ 5	⊗ Non	⊗ Nor *

✓ Prise en charge complète ⊗ Pas de prise en charge * Voir les notes de mise en application.

Voir aussi

- [Envoyer des notifications WebPush identifiées avec VAPID grâce au service Push de Mozilla \(en anglais\)](#) ↗
- [Démon API Push sur GitHub](#) ↗
- [Aperçu général des notifications *push* \(en anglais\)](#) ↗
- [L'API *Service Worker*](#)



Votre modèle pour un internet meilleur.