Options of a PushSubscription

By Matt Gaunt

Matt is a contributor to Web**Fundamentals**

When a **`pushsubscriptionchange`** event occurs, it's an opportunity for a developer to re-subscribe the user for push. One of the pain points of this is that to re-subscribe a user, the developer has to keep the **`applicationServerKey`** (and any other **`subscribe()`** options) in sync between the web page's JavaScript and their service worker.

In Chrome 54 and later you can now access the options via the options parameter in a subscription object, known as PushSubscriptionOptions.

You can copy and paste the following code snippet into simple-push-demo to see what the options look like. The code simply gets the current subscription and prints out **`subscription.options`**.

```
navigator.serviceWorker.ready.then(registration => {
  return registration.pushManager.getSubscription();
})
.then(subscription => {
  if (!subscription) {
    console.log('No subscription ☹');
    return;
  }

  console.log('Here are the options 🎉');
  console.log(subscription.options);
});
```

With this small piece of information you can re-subscribe a user in the pushsubscriptionchange event like so:

```
self.addEventListener('pushsubscriptionchange', e => {
  e.waitUntil(registration.pushManager.subscribe(e.oldSubscription.options)
    .then(subscription => {
      // TODO: Send new subscription to application server
    }));
});
```

It's a small change, that will be super useful in the future.

Except as otherwise noted, the content of this page is licensed under the Creative Commons Attribution 3.0 License, and code samples are licensed under the Apache 2.0 License. For details, see our Site Policies. Java is a registered trademark of Oracle and/or its affiliates.

Last updated July 2, 2018.