Ablauf Umsetzungsvorstellung

Zuerst muss der Nutzer der Web-App auf eine Anforderung für Webbenachrichtigungen oder sonstige verwendete Berechtigungen reagieren, indem er der App die Berechtigungen erteilt (**siehe Bild**).

Nachdem die Berechtigung erfolgt ist, wird der Service-Worker, lokal für die Web-Anwendung registriert.

Danach wird der Push-Messaging-Service, in unserem Fall „Firebase Cloud Messaging“ kurz FCM mit der Funktion PushManager.subscribe() abonniert.

Daraufhin wird mit der Funktion PushSubscription.endpoint () der mit der Subscription verknüpfte Endpoint aufgerufen.

Die Details werden an den Application-Server gesendet, so dass er Push-Nachrichten senden kann, wenn dies erforderlich ist. Die Subscription-ID wird aus dem kompletten Endpoint entnommen.

Auf der Serverseite wird der Endpoint und alle anderen erforderlichen Details, wie die Sender ID und Geräte ID in der MongoDB gespeichert, so dass sie verfügbar sind, wenn eine Push-Nachricht an einen Push-Absender gesendet werden soll.

Um eine Push-Nachricht zu senden, muss ein HTTP-POST an die Endpoint-URL gesendet werden. Die Anforderung muss einen TTL-Header enthalten, der begrenzt, wie lange die Nachricht in der Warteschlange stehen soll, wenn der Benutzer nicht online ist.

Um Nutzdaten in Ihre Anfrage einzubinden, müssen Sie verschlüsselt werden (mit dem öffentlichen Schlüssel des Clients (public key)).

Da wir uns gegen eine Nutzdatensendung über den Browseranbieter entschieden haben, entfällt bei uns dieser Schritt mit der Vergabe von Public und Primary Key .

Der Web-Push-Server antwortet mit Response und sobald Push Nachrichten erfolgreich gesendet wurden, dann enthält dieser die Eindeutigen Message ID´s (eine Referenz auf verschickte Push-Nachricht).

Der Application-Server ordnet dann der Message ID den zuvor generierten Payload (Nutzdaten) zu und hält diese für Clientanfragen vor.

Sobald nun eine Push Nachricht an den Client übertragen wird, löst dieses Event ein weiteres onPush Event aus, daraufhin wird der vorgehaltene Payload direkt von unserem Applikation Server an den Client übertragen. Dies ist der Grund, warum bei uns die Verschlüsselung entfällt.