

بخش سوم از سوال ۳ :

چرا caop و mqtt جایگزین http میشوند؟

### : Coap vs HTTP

برنامه صنعت این است که HTTP را با انتقال به HTTP / 2 بهبود ببخشید ، و HTTP / 2 شامل (در میان سایر ویژگی ها) فشرده سازی هدر است ، که مزایای مشابه CoAP را برای شما به همراه دارد.

در حالی که امروزه بیشتر وب سرورها و برخی مرورگرها از HTTP / 2 پشتیبانی می کنند ، AFAIK هیچ مرورگری و نه هیچ سروری از CoAP پشتیبانی نمی کند.

Coap برای resource constrained network ها طراحی شده است و امروزه به خاطر این کاربرد به سرعت در حال توسعه است coap کاربرد بیشتری نسبت به http دارد.

### : HTTP vs MQTT

در مورد MQTT در مقابل HTTP ادبیات غنی وجود دارد که باید برای جزئیات بیشتر به آن پردازید ، اما در اینجا برخی از نکات مهم در مورد برنامه شما ذکر شده است..

### MQTT

MQTT امکان اتصال مداوم را فراهم می کند که می تواند منابع قابل توجهی را از طریق HTTP صرفه جویی کند. این موضوع در صورتی که از SSL استفاده می کنید بیشترین اهمیت را دارد.

اگر فقط چند معیار ارسال می کنید ، پهنای باند MQTT نسبت به HTTP به طور کلی کارآمدتر است.

همانطور که MQTT برای ارائه داده ها (نه کل صفحات) طراحی شده است ، مدل فرعی آن بسیاری از ویژگی های داخلی داخلی مانند نگهداری و آخرین اراده را ارائه می دهد.

همچنین MQTT یک روش ساده برای پیاده سازی رمزگذاری ، احراز هویت و کنترل دسترسی را ارائه می دهد.

MQTT برای موقعیت هایی در نظر گرفته شده است که اتصال شما ممکن است متناوب باشد یا در غیر این صورت غیر قابل اعتماد باشد. سطوح مختلف کیفیت خدمات آن روش های قابل توجهی را برای اطمینان از ارسال مطمئن داده ها به شما ارائه می دهد.

به طور خلاصه ، MQTT دارای ویژگی های بسیاری است که به نظر می رسد برای برنامه شما مناسب است. با این گفته ، شما احتمالاً می توانید بسیاری از آنها را با استفاده از HTTP تکرار کنید اما به کار بیشتری نیاز دارد.

## HTTP

تقریباً همه جا پشتیبانی می شود که سازگاری راحت تری با فایروال ها را تضمین می کند. اگر در شبکه هایی که کنترل نمی کنید مستقر می شوید ، این مهم می تواند باشد.

این یک پروتکل رایج است بنابراین شما (و کاربران نهایی خود) ممکن است از قبل با آن احساس راحتی کنید. به همین ترتیب شما ممکن است از قبل مدل امنیتی را درک کنید که امنیت آن را آسان تر می کند.

تفاوت هایی بین پیاده سازی های مختلف MQTT وجود دارد که ممکن است مشکلاتی را ایجاد کند (به عنوان مثال من از Mosquitto استفاده می کنم و گاهی اوقات گیج کننده می دانم وقتی مردم در مورد تنظیمات HiveMQ صحبت می کنند). HTTP جهانی تر به نظر من می رسد و جامعه بسیار گسترده تری از افراد برای کمک به شما در دسترس است.

بنابراین HTTP در مقابل MQTT دارای معایب ذاتی است اما اگر ویژگی های خاص MQTT به نظر شما جذاب نباشد کار را انجام می دهد و عملی تر است.

اما واقعاً اگر این یک پروژه در مقیاس بزرگ است ، ارزش دارد هر دو را امتحان کنید و برخی از معیارها و تست ها را در برنامه و محیط خاص خود اجرا کنید. راه اندازی یک محیط آزمون و دریافت برخی از معیارها کار سختی نخواهد بود. اگر این بیشتر یک پروژه سرگرمی / یکبار مصرف باشد ، من از هر کدام با راحت تر یا جالب تر استفاده می کنم.

مزایای mqtt بر http:

Mqtt میتواند اتصال های پایدار ایجاد کند که در نتیجه از منابع کمتری نسبت به http استفاده می کند.

Mqtt برای فرستادن داده های با تعداد کمتر از لحاظ پهنای داده بهینه تر است.

Mqtt روشی آسان برای پیاده سازی رمز گذاری و کنترل دسترسی ارائه می دهد.

Mqtt برای موقعیت هایی که از اتصالات متناوب استفاده می شود کاربرد دارد و دارای level های مختلفی برای کیفیت سرویس ها برای اطمینان از ارسال بدون خطای داده است.

ساختار topic/subtopic کمک میکند که داده براحتی سازماندهی شود و منابع به صورت مناسب بین پروژه های مختلف تقسیم شود.