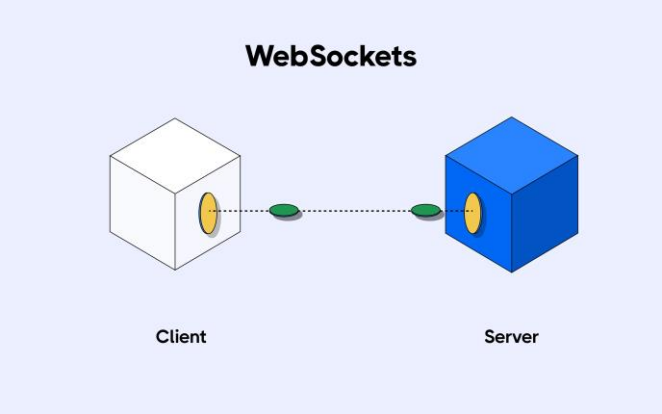
WebSockets e Webhooks: Uma Visão Técnica

WebSockets

O que são: WebSockets são um protocolo de comunicação em tempo real que permite a comunicação bidirecional entre um cliente e um servidor. Eles mantêm uma conexão persistente aberta, permitindo que dados sejam enviados e recebidos continuamente sem a necessidade de múltiplas requisições HTTP.



Como Funcionam

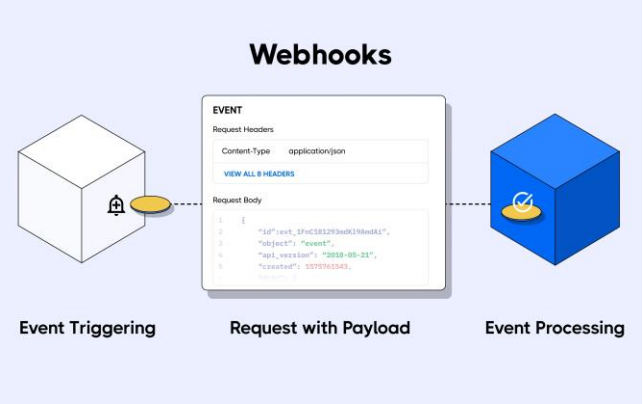
A conexão WebSocket é estabelecida através de uma negociação de handshake HTTP. Após o handshake, a conexão é atualizada para o protocolo WebSocket, onde o cliente e o servidor podem trocar mensagens através de quadros de dados. Esses quadros possuem um cabeçalho com informações como o comprimento e o tipo de mensagem, além do payload, que é a própria mensagem.

Segurança

- WSS (WebSockets sobre SSL/TLS): Utilizar wss:// em vez de ws:// garante que a conexão seja criptografada, protegendo contra ataques do tipo man-in-the-middle.
- Validação de entrada e saída: Tanto os dados do cliente quanto os do servidor devem ser validados para evitar injeções SQL, XSS, e outros tipos de ataques.
- Controle de acesso: Implementar verificações de controle de acesso para garantir que os usuários só possam acessar recursos autorizados.
- Autenticação: A autenticação deve ser configurada explicitamente, já que WebSockets não gerenciam isso automaticamente. Utilizar tokens ou outros mecanismos de autenticação.
- Evitar túnel de serviços: Evitar usar WebSockets para tunelamento de serviços como conexões de banco de dados, pois isso pode expor serviços críticos a ataques.

Webhooks

O que são: Webhooks são chamadas de retorno HTTP que permitem que um sistema envie dados para outro sistema quando um evento específico ocorre. Eles são frequentemente usados para integrar diferentes serviços e automatizar processos.



Como Funcionam

Quando um evento ocorre no sistema de origem, o Webhook envia uma solicitação HTTP POST para uma URL configurada no sistema de destino. O payload dessa solicitação contém informações sobre o evento.

Segurança

- Autenticação: Utilizar tokens de autenticação para verificar a origem das solicitações.
- Validação de dados: Validar os dados recebidos para evitar injeções e ataques de segurança.
- HTTPS: Sempre utilizar HTTPS para garantir que os dados transmitidos estejam criptografados.
- Restrição de IP: Restringir as solicitações de Webhook a endereços IP confiáveis para evitar acessos não autorizados.

Comparação

Comunicação

WebSockets: Comunicação bidirecional em tempo real, ideal para aplicações que exigem troca contínua de dados, como chats, jogos online e dashboards em tempo real.

Webhooks: Funcionam de forma unidirecional, onde um sistema envia dados para outro apenas quando um evento específico ocorre.

Persistência

WebSockets: Mantêm uma conexão persistente aberta, adequada para interações contínuas e em tempo real.
Webhooks: Não mantêm uma conexão aberta; cada evento gera uma nova solicitação HTTP.

Complexidade de Implementação

WebSockets: Podem aumentar a complexidade do servidor, pois requerem a manutenção de múltiplas conexões abertas.

Webhooks: São mais simples de implementar e não requerem a manutenção de conexões persistentes.

Uso de Recursos

WebSockets: Podem ser mais eficientes em termos de uso de recursos para aplicações que requerem comunicação continua.
Webhooks: São mais eficientes para eventos esporádicos, economizando recursos ao não manter conexões abertas.

Casos de Uso

WebSockets: Aplicações de chat, jogos multiplayer, atualizações em tempo real (como preços de ações), e colaboração em tempo real.
Webhooks: Integrações entre serviços, notificações de eventos (como novos commits em um repositório de código), e automação de workflows.

Fontes

- Ably Realtime — WebSocket API and protocol explained. Disponível em: https://www.ramotion.com/blog/what-is-websocket/
- The Knowledge Academy — What is WebSocket and What are they Used for? Disponível em: https://www.theknowledgeacademy.com/blog/what-is-websocket/
- Sookocheff, Kevin. How Do Websockets Work? Disponível em: https://sookocheff.com/post/networking/how-do-websockets-work/
- Diffusion Data — What are WebSockets and how do they work? Disponível em: https://www.diffusiondata.com/what-are-web-sockets-and-how-do-they-work/
- WebSocket security: How to prevent 9 common vulnerabilities, Booker, Alex. Disponível em: https://ably.com/topic/websocket-security
- WebSocket Security, Heroku Dev Center. Disponível em: https://devcenter.heroku.com/articles/websocket-security