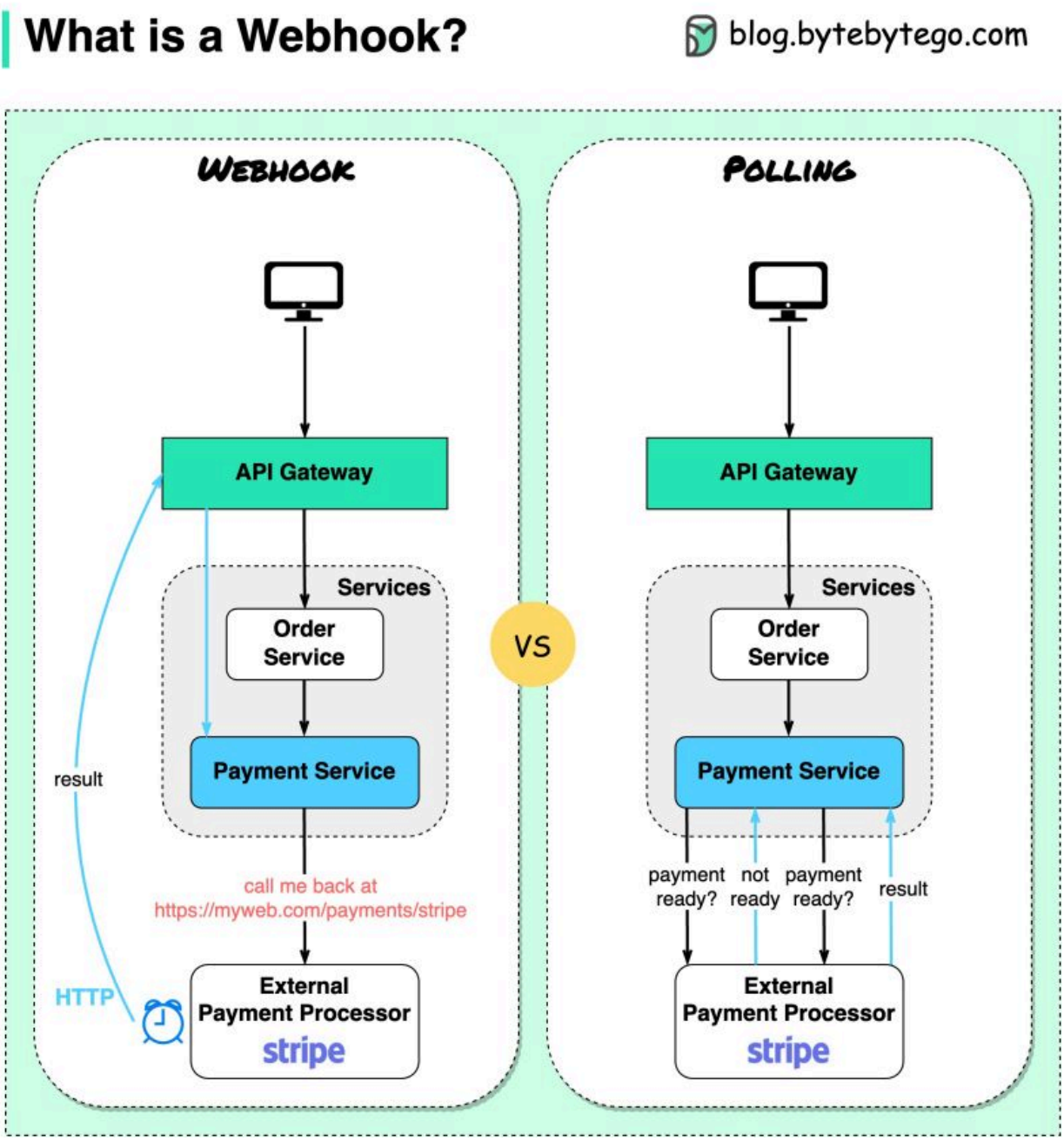
What is a webhook?

The diagram below shows a comparison between polling and webhook.



Assume we run an eCommerce website. The clients send orders to the order service via the API gateway, which goes to the payment service for payment transactions. The payment service then talks to an external payment service provider (PSP) to complete the transactions.

There are two ways to handle communications with the external PSP.

1. Short polling
   After sending the payment request to the PSP, the payment service keeps asking the PSP about the payment status. After several rounds, the PSP finally returns with the status.

   Short polling has two drawbacks:
   - Constant polling of the status requires resources from the payment service.
   - The External service communicates directly with the payment service, creating security vulnerabilities.

2. Webhook
   We can register a webhook with the external service. It means: call me back at a certain URL when you have updates on the request. When the PSP has completed the processing, it will invoke the HTTP request to update the payment status.

   In this way, the programming paradigm is changed, and the payment service doesn't need to waste resources to poll the payment status anymore.

What if the PSP never calls back? We can set up a housekeeping job to check payment status every hour.

Webhooks are often referred to as reverse APIs or push APIs because the server sends HTTP requests to the client. We need to pay attention to 3 things when using a webhook:

1. We need to design a proper API for the external service to call.
2. We need to set up proper rules in the API gateway for security reasons.
3. We need to register the correct URL at the external service.