# Routing: Apache httpd

In this lesson, we'll discuss routing with Apache httpd.

WE'LL COVER THE FOLLOWING



## Introduction

The Apache httpd server is one of the most widely used web servers. There are modules that adapt the server to different usage scenarios. In the example, modules are configured that turn Apache httpd into a reverse proxy.

## Reverse proxy

While a **conventional proxy** can be used to process traffic from a **network to the outside**, a **reverse proxy** is a solution for **inbound network** connections.

It can forward external requests to specific services. This means that the entire microservices system can be accessible under one URL but can use different microservices internally.

The concept of a reverse proxy has already been explained in the lesson, Router: Zuul.

## Load balancer

In addition, Apache httpd serves as a **load balancer** by distributing network traffic to make the application scalable.

In the example, there is only one Apache httpd, that functions simultaneously as a reverse proxy and a load balancer for requests from the outside.

The requests the microservices send to each other are not handled by this load balancer. For the communication between the microservices, the library **Ribbon** is used, as we already saw in the Netflix example (see Load Balancing: Ribbon).

One of the strengths of this solution is that it uses a **well-proven software** that many teams have already gained experience with.

As microservices place high demands on the operation and infrastructure, such a conservative choice is advantageous to avoid the effort that goes into learning another technology. Instead of Apache httpd you can also use example nginx.

There are also approaches like Fabio which are written specifically for the load balancing of microservices and are easier to use and configure.

## QUIZ

1 Can Apache httpd work as a load balancer and reverse proxy at the same time?

COMPLETED 0%

1 of 3

In the next lesson, we'll discuss the Consul template.