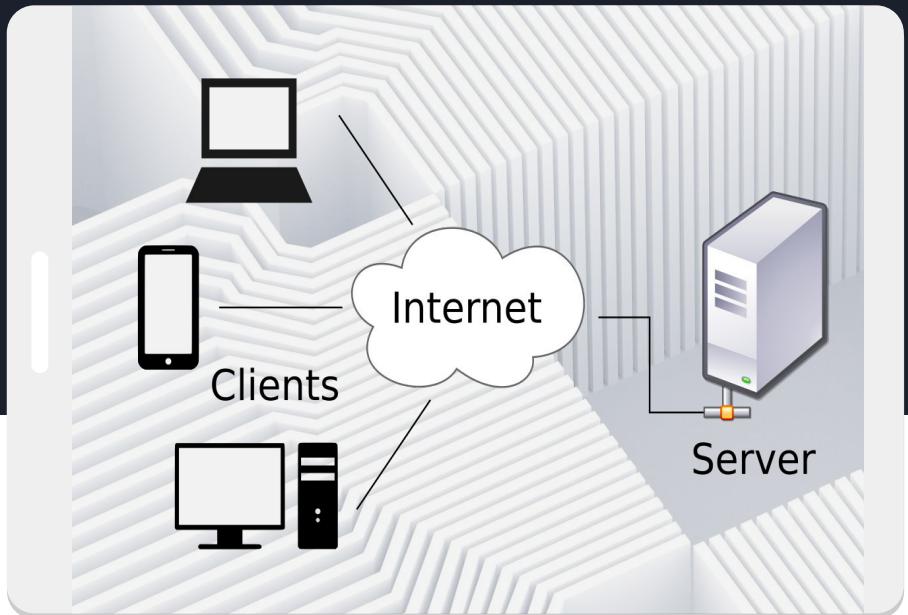


TCP Client-Server: The Building Block of the Internet

Workshop on how to build a simple TCP server.
SBy: Shad Beard and Will Hawkins

What is the Client-Server Model?

A model of communication in which a user (client) makes a request from a service (server) and the service responds (often in the form of information).





What Are Some Things That Use TCP

01

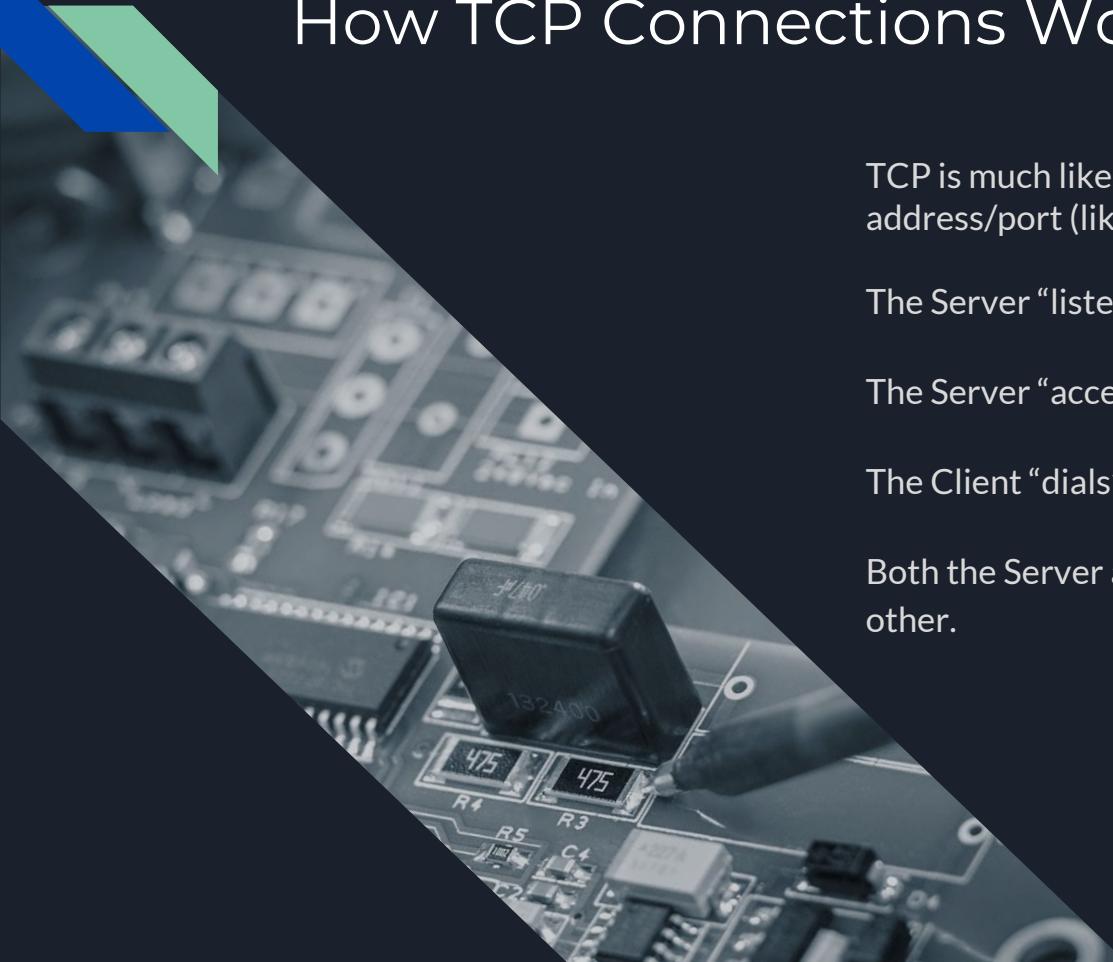
HTTP(S): This is the protocol that the Internet is built on. It is a layer on top of TCP. Browsers use HTTP to display websites and receive, update, create, and delete data from the Internet.

02

Databases: The mechanism for sending information to a database uses TCP. Common databases that use TCP include MySQL, MongoDB, Redis, and many others.

03

Connected Devices: Many of the protocols for devices used in robotics and manufacturing are built on top of TCP



How TCP Connections Work

TCP is much like a telephone call. It consists of an address/port (like a phone number).

The Server “listens” for incoming requests (similar to a phone).

The Server “accepts” an incoming request.

The Client “dials” into the server to establish a “connection”.

Both the Server and Client read and write messages to each other.

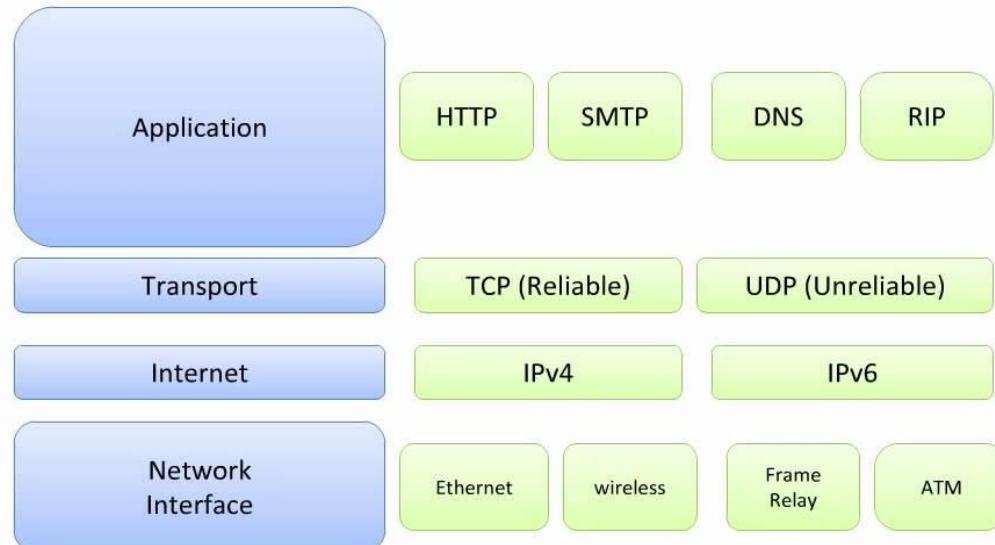
TCP/IP and OSI Models

OSI and TCP/IP Model

OSI Model Layers

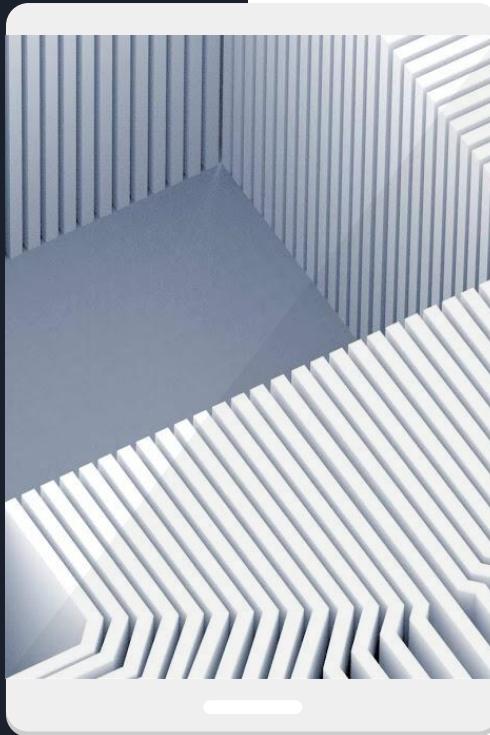


TCP/IP Model Layers



TCP Client & Server in Go

What we will be building in this workshop



We will be walking through a simple TCP client and server. We will be using the Go programming language. In order to make it easier to control the environment, you will ssh into a VM hosted on AWS (Don't worry, we'll show you how to do that).

Let's Get Started

<https://github.com/Dahs81/tech-olympics-demo>

