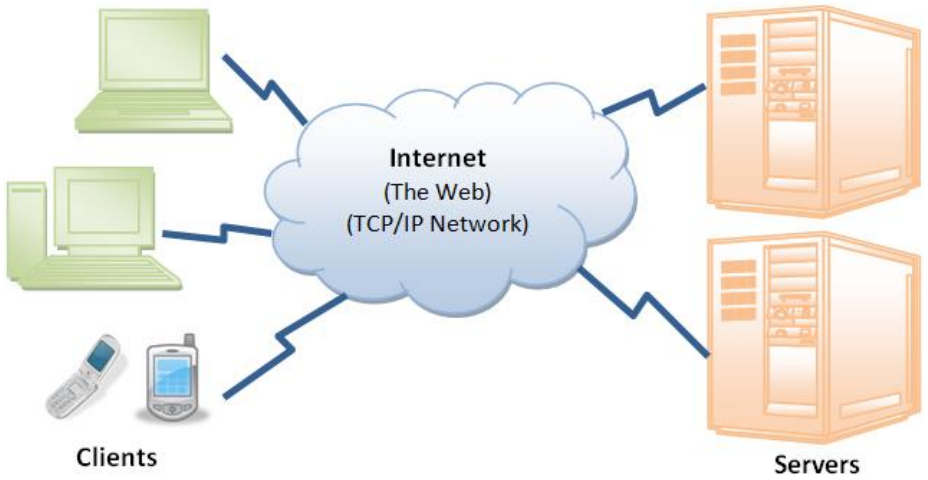
Qt:n tietoliikenneohjelmointi

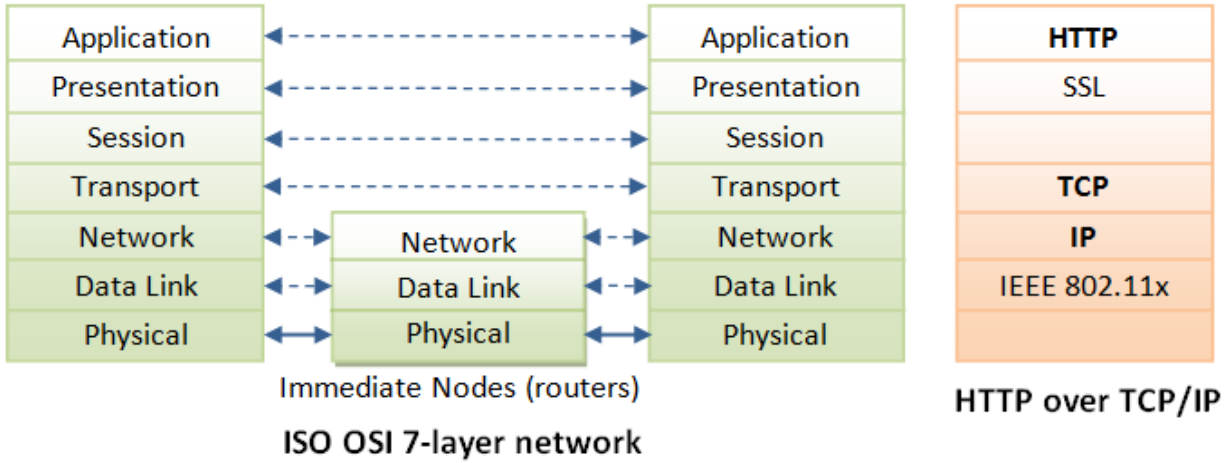
The WEB

- Internet (or The Web) is a massive distributed client/server information system as depicted in the following diagram.

- Many applications are running concurrently over the Web, such as web browsing/surfing, e-mail, file transfer, audio & video streaming, and so on. In order for proper communication to take place between the client and the server, these applications must agree on a specific application-level protocol such as HTTP, FTP, SMTP, POP, and etc.



TCP/IP (Transmission Control Protocol/Internet Protocol)



- TCP/IP is a set of transport and network-layer protocols for machines to communicate with each other over the network.
- Stream Sockets

- Stream sockets use a **Transmission Control Protocol (TCP)**, which exists on the transport layer of the Open Systems Interconnection (OSI) model. The data is usually transmitted in packets. TCP is designed so that the packets of data will arrive without errors and in sequence. Webservers, mail servers, and their respective client applications all use TCP and stream socket to communicate.

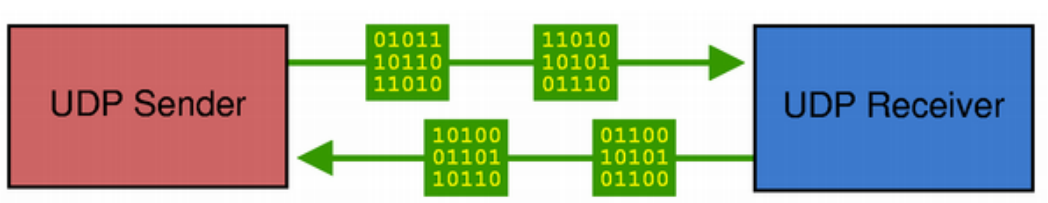
- Stream sockets provide **reliable two-way** communication similar to when we call someone on the phone. One side initiates the connection to the other, and after the connection is established, either side can communicate to the other. In addition, there is immediate confirmation that what we said actually reached its destination.

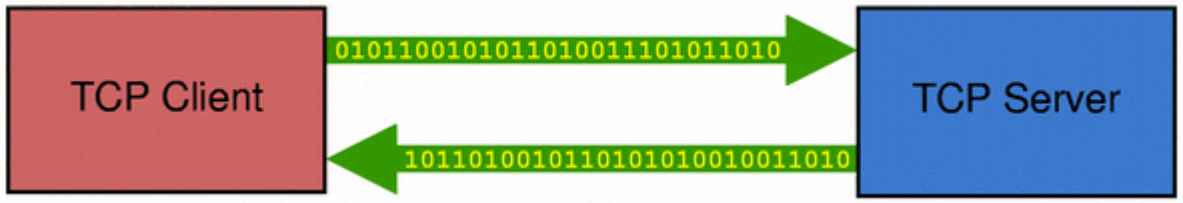
TCP/IP (Transmission Control Protocol/Internet Protocol)

- IP (Internet Protocol) is a network-layer protocol, deals with network addressing and routing. In an IP network, each machine is assigned an unique IP address (e.g., 165.1.2.3), and the IP software is responsible for routing a message from the source IP to the destination IP. In IPv4 (IP version 4), the IP address consists of 4 bytes, each ranges from 0 to 255, separated by dots, which is called a quad-dotted form. This numbering scheme supports up to 4G addresses on the network. The latest IPv6 (IP version 6) supports more addresses. Since memorizing number is difficult for most of the people, an english-like domain name, such as www.nowhere123.com is used instead. The DNS (Domain Name Service) translates the domain name into the IP address (via distributed lookup tables). A special IP address 127.0.0.1 always refers to your own machine. It's domian name is "localhost" and can be used for local loopback testing.

- TCP (Transmission Control Protocol) is a transport-layer protocol, responsible for establish a connection between two machines. TCP consists of 2 protocols: TCP and UDP (User Datagram Package). TCP is reliable, each packet has a sequence number, and an acknowledgement is expected. A packet will be re-transmitted if it is not received by the receiver. Packet delivery is guaranteed in TCP. UDP does not guarantee packet delivery, and is therefore not reliable. However, UDP has less network overhead and can be used for applications such as video and audio streaming, where reliability is not critical.

- TCP multiplexes applications within an IP machine. For each IP machine, TCP supports (multiplexes) up to 65536 ports (or sockets), from port number 0 to 65535. An application, such as HTTP or FTP, runs (or listens) at a particular port number for incoming requests. Port 0 to 1023 are pre-assigned to popular protocols, e.g., HTTP at 80, FTP at 21, Telnet at 23, SMTP at 25, NNTP at 119, and DNS at 53. Port 1024 and above are available to the users.





A Network Socket

- A network socket is an internal endpoint for sending or receiving data at a single node in a computer network.
- The term "socket" is analogous to physical female connectors, communication between two nodes through a channel being visualized as a cable with two male connectors plugging into sockets at each node.
- A socket can be referred to by a process (a running computer program) by using a **socket descriptor**, a type of handle (abstract reference, often represented internally as an integer).
- In practice "socket" usually refers to a socket in an Internet Protocol (IP) network (where sockets may be called Internet sockets), in particular for the Transmission Control Protocol (TCP), which is a protocol for one-to-one connections. In this context, sockets are assumed to be associated with a specific socket address, namely the IP address and a port number for the local node.
- A protocol stack, today usually provided by the operating system (rather than as a separate library, for instance), is a set of programs that allow processes to communicate over a network using the protocols that the stack implements. The application programming interface (API) that programs use to communicate with the protocol stack, using network sockets, is called a socket API. Development of application programs that utilize this API is called socket programming or network programming.
- In the standard Internet protocols TCP and UDP, a socket address is the combination of an IP address and a port number, much like one end of a telephone connection is the combination of a phone number and a particular extension. Sockets need not have an address, for example, for only sending data, but if a program binds a socket to an address, the socket can be used to receive data sent to that address. Based on this address, Internet sockets deliver incoming data packets to the appropriate application process.

A Network Socket, here is the summary of key concepts:

1. Socket is a way of speaking to other programs using standard **file descriptors**.
2. Where do we get the file descriptor for network communication?
   - Well, we make a call to the **socket()** system routine.
   - After the socket() returns the socket descriptor, we start communicate through it using thespecialized **send()/recv()** socket API calls.

3.A TCP socket is an **endpoint instance**

4.A TCP socket is **not a connection**, it is the **endpoint** of a specific connection.

5.A TCP **connection** is defined by **two endpoints** aka sockets.

6.The purpose of **ports** is to **differentiate multiple endpoints** on a given network address.

7.The port numbers are encoded in the transport protocol packet header, and they can be readily interpreted not only by the sending and receiving computers, but also by other components of the networking infrastructure. In particular, firewalls are commonly configured to differentiate between packets based on their source or destination **port numbers** as in **port forwarding**.

8.It is the **socket pair** (the **4-tuple** consisting of the client IP address, client port number, server IP address, and server port number) that specifies the two endpoints that uniquely identifies each **TCP connection** in an internet.

9.Only **one process** may bind to a specific **IP address** and **port** combination using the **same transport protocol**. Otherwise, we'll have **port conflicts**, where multiple programs attempt to bind to the same port numbers on the same IP address using the same protocol.

To connect to another machine, we need a **socket** connection.

What’s a connection?

- To connect to another machine, we need a **socket** connection.

- What's a connection?

- A **relationship** between two machines, where **two pieces of software know about each other**. Those two pieces of software know how to communicate with each other. In other words, they know how to send **bits** to each other.
A socket connection means the two machines have information about each other, including **network location (IP address)** and **TCP port**. (If we can use anology, IP address is the **phone number** and the TCP port is the **extension**).
A socket is an object similar to a file that allows a program to accept incoming connections, make outgoing connections, and send and receive data. **Before two machines can communicate, both must create a socket object**.
A socket is a **resource** assigned to the server process. The server creates it using the system call **socket()**, and it can't be shared with other processes.

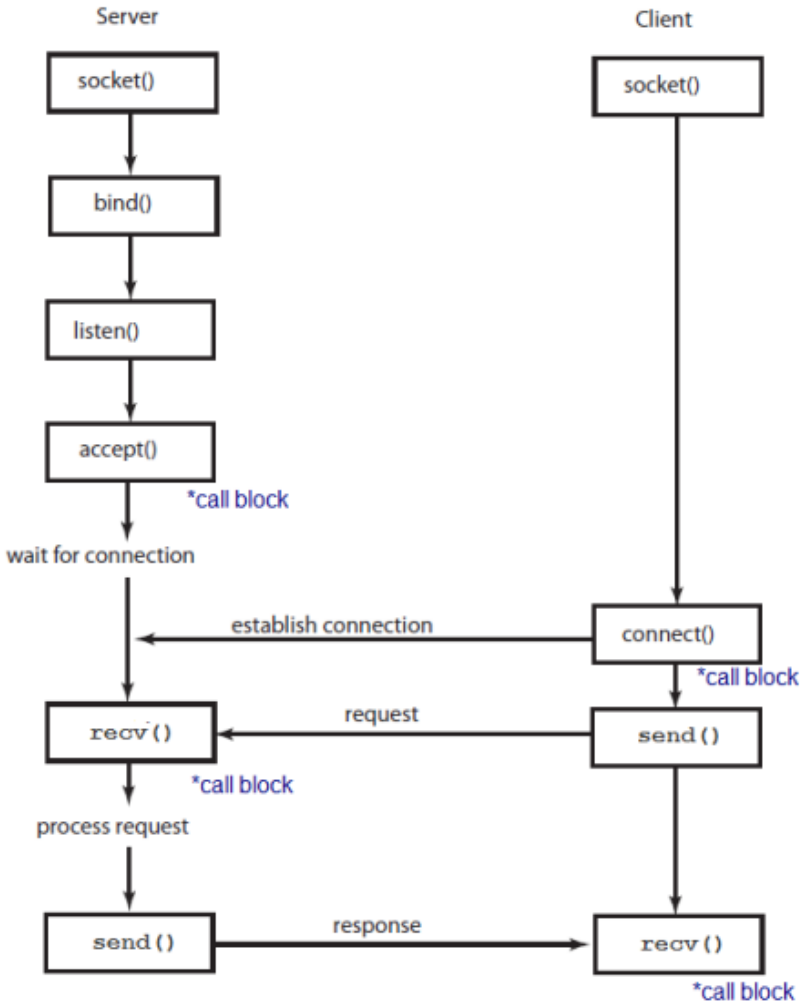
Client/Server

- The client-server model distinguishes between applications as well as devices. Network clients make requests to a server by sending messages, and servers respond to their clients by acting on each request and returning results.

- For example, let's talk about telnet.
  When we connect to a remote host on port 23 with telnet (the client), a program on that host (called telnetd, the server) springs to life. It handles the incoming telnet connection, sets us up with a login prompt, etc.

- One server generally supports numerous clients, and multiple servers can be networked together in a pool to handle the increased processing load as the number of clients grows.
  Some of the most popular applications on the Internet follow the client-server model including email, FTP and Web services. Each of these clients features a user interface and a client application that allows the user to connect to servers. In the case of email and FTP, users enter a computer name (or an IP address) into the interface to set up connections to the server.

TCP Socket, Client and Server communication




```
socket = new QTcpSocket();
```

```
socket->connectToHost(”oamk.fi",80);
```

```
socket->write("Hello server\r\n\r\n");
```

```
socket->readAll();
```