

# Networking, Part 1: Introduction

nickgarfield edited this page on Nov 30, 2014 · 8 revisions

Caveat: It should be obvious that page is *not* a complete description of IP, UDP or TCP! Instead it is a short introduction and is sufficient so that we can build upon these concepts in later lectures.

## What is "IP4" "IP6"?

The following is the "30 second" introduction to internet protocol (IP) - which is the primary way to send packets ("datagrams") of information from one machine to another.

"IP4", or more precisely, "IPv4" is version 4 of the Internet Protocol that describes how to send packets of information across a network from one machine to another . Roughly 95% of all packets on the Internet today are IPv4 packets. A significant limitation of IPv4 is that source and destination addresses are limited to 32 bits (IPv4 was designed at a time when the idea of 4 billion devices connected to the same network was unthinkable - or at least not worth making the packet size larger)

Each IPv4 packet includes a very small header - typically 20 bytes (more precisely, "octets"), that includes a source and destination address.

Conceptually the source and destination addresses can be split into two: a network number (the upper bits) and the lower bits represent a particular host number on that network.

A newer packet protocol "IPv6" solves many of the limitations of IPv4 (e.g. makes routing tables simpler and 128 bit addresses) however less than 5% of web traffic is IPv6 based.

A machine can have an IPv6 address and an IPv4 address.

## "There's no place like 127.0.0.1"

A special IPv4 address is `127.0.0.1` also known as localhost. Packets sent to 127.0.0.1 will never leave the machine; the address is specified to be the same machine.

Notice that the 32 bits address is split into 4 octets i.e. each number in the dot notation can be 0-255 inclusive. However IPv4 addresses can also be written as an integer.

## ... and ... "There's no place like 0:0:0:0:0:0:0:1?"

The 128bit localhost address in IPv6 is `0:0:0:0:0:0:0:1` which can be written in its shortened form, `::1`

Edit

New Page

▼ Pages 51

Home

#Example Markdown

#Informal Glossary

#Piazza: When And How to Ask For Help

C Programming, Part 1: Introduction

C Programming, Part 2: Text Input And Output

C Programming, Part 3: Common Gotchas

C Programming, Part 4: Debugging

Deadlock, Part 1: Resource Allocation Graph

Deadlock, Part 2: Deadlock Conditions

File System, Part 1: Introduction

File System, Part 2: Files are inodes (everything else is just data...)


File System, Part 3: Permissions

File System, Part 4: Working with directories

File System, Part 5: Virtual file systems

Show 36 more pages...

Clone this wiki locally

 Clone in Desktop

# What is a port?

To send a datagram (packet) to a host on the Internet using IPv4 (or IPv6) you need to specify the host address and a port. The port is an unsigned 16 bit number (i.e. the maximum port number is 65535).

A process can listen for incoming packets on a particular port. However only processes with super-user (root) access can listen on ports < 1024. Any process can listen on ports 1024 or higher.

An often used port is port 80: Port 80 is used for unencrypted http requests (i.e. web pages). For example, if a web browser connects to <http://www.bbc.com/>, then it will be connecting to port 80.

# What is UDP? When is it used?

UDP is a connectionless protocol that is built on top of IPv4 and IPv6. It's very simple to use: Decide the destination address and port and send your data packet! However the network makes no guarantee about whether the packets will arrive. Packets (aka Datagrams) may be dropped if the network is congested. Packets may be duplicated or arrive out of order.

Between two distant data-centers it's typical to see 3% packet loss.

A typical use case for UDP is when receiving up to date data is more important than receiving all of the data. For example, a game may send continuous updates of player positions. A streaming video signal may send picture updates using UDP

# What is TCP? When is it used?

TCP is a connection-based protocol that is built on top of IPv4 and IPv6 (and therefore can be described as "TCP/IP" or "TCP over IP"). TCP creates a *pipe* between two machines and abstracts away the low level packet-nature of the Internet: Thus, under most conditions, bytes sent from one machine will eventually arrive at the other end without duplication or data loss.

TCP will automatically manage resending packets, ignoring duplicate packets, re-arranging out-of-order packets and changing the rate at which packets are sent.

Most services on the Internet today (e.g. a web service) use TCP because it hides the complexity of lower, packet-level nature of the Internet.

Legal and Licensing information: Unless otherwise specified, submitted content to the wiki must be original work (including text, java code, and media) and you provide this material under a [Creative Commons License](#). If you are not the copyright holder, please give proper attribution and credit to existing content and ensure that you have license to include the materials.

