# Networking Glossary

First, we will define some common terms that you will see throughout this guide, and in other guides and documentation regarding networking.

These terms will be expanded upon in the appropriate sections that follow:

- Connection: In networking, a connection refers to pieces of related information that are transferred through a network. Generally speaking, a connection is established before data transfer (by following the procedures laid out in a protocol) and may be deconstructed at the end of the data transfer.
- Packet: A packet is the smallest unit that is intentionally transferred over a network. When communicating over a network, packets are the envelopes that carry your data (in pieces) from one end point to the other.

Packets have a header portion that contains information about the packet including the source and destination, timestamps, network hops, etc. The main portion of a packet contains the actual data being transferred. It is sometimes called the body or the payload.

- Network Interface: A network interface can refer to any kind of software interface to networking hardware. For instance, if you have two network cards in your computer, you can control and configure each network interface associated with them individually.

A network interface may be associated with a physical device, or it may be a representation of a virtual interface. The “loopback” device, which is a virtual interface available in most Linux environments to connect back to the same machine, is an example of this.

- LAN: LAN stands for “local area network”. It refers to a network or a portion of a network that is not publicly accessible to the greater internet. A home or office network is an example of a LAN.
- WAN: WAN stands for “wide area network”. It means a network that is much more extensive than a LAN. While WAN is the relevant term to use to describe large, dispersed networks in general, it is usually meant to mean the internet, as a whole.

If an interface is said to be connected to the WAN, it is generally assumed that it is reachable through the internet.

- Protocol: A protocol is a set of rules and standards that define a language that devices can use to communicate. There are a great number of protocols in use extensively in networking, and they are often implemented in different layers.

Some low level protocols are TCP, UDP, IP, and ICMP. Some familiar examples of application layer protocols, built on these lower protocols, are HTTP (for accessing web content), SSH, and TLS/SSL.

- Port: A port is an address on a single machine that can be tied to a specific piece of software. It is not a physical interface or location, but it allows your server to be able to communicate using more than one application.
- Firewall: A firewall is a program that decides whether traffic coming or going from a server should be allowed. A firewall usually works by creating rules for which type of traffic is acceptable on which ports. Generally, firewalls block ports that are not used by a specific application on a server.
- NAT: NAT stands for network address translation. It is a way to repackage and send incoming requests to a routing server to the relevant devices or servers on a LAN. This is usually implemented in physical LANs as a way to route requests through one IP address to the necessary backend servers.
- VPN: VPN stands for virtual private network. It is a means of connecting separate LANs through the internet, while maintaining privacy. This is used to connect remote systems as if they were on a local network, often for security reasons.

There are many other terms that you will come across, and this list is not exhaustive. We will explain other terms as we need them. At this point, you should understand some high-level concepts that will enable us to better discuss the topics to come.

## Network Layers

While networking is often discussed in terms of topology in a horizontal way, between hosts, its implementation is layered in a vertical fashion within any given computer or network.

What this means is that there are multiple technologies and protocols that are built on top of each other in order for communication to function. Each successive, higher layer abstracts the raw data a little bit more.

It also allows you to leverage lower layers in new ways without having to invest the time and energy to develop the protocols and applications that handle those types of traffic.

The language that we use to talk about each of the layering schemes varies significantly depending on which model you use. Regardless of the model used to discuss the layers, the path of data is the same.

As data is sent out of one machine, it begins at the top of the stack and filters downwards. At the lowest level, actual transmission to another machine takes place. At this point, the data travels back up through the layers of the other computer.

Each layer has the ability to add its own “wrapper” around the data that it receives from the adjacent layer, which will help the layers that come after decide what to do with the data when it is handed off.

### TCP/IP Model

The TCP/IP model, more commonly known as the Internet protocol suite, is a widely adopted layering model. It defines the four separate layers:

- Application: In this model, the application layer is responsible for creating and transmitting user data between applications. The applications can be on remote systems, and should appear to operate as if locally to the end user. This communication is said to take place between peers.
- Transport: The transport layer is responsible for communication between processes. This level of networking utilizes ports to address different services.
- Internet: The internet layer is used to transport data from node to node in a network. This layer is aware of the endpoints of the connections, but is not concerned with the actual connection needed to get from one place to another. IP addresses are defined in this layer as a way of reaching remote systems in an addressable manner.
- Link: The link layer implements the actual topology of the local network that allows the internet layer to present an addressable interface. It establishes connections between neighboring nodes to send data.

As you can see, the TCP/IP model is abstract and fluid. This made it popular to implement and allowed it to become the dominant way that networking layers are categorised.

## Interfaces

Interfaces are networking communication points for your computer. Each interface is associated with a physical or virtual networking device.

Typically, your server will have one configurable network interface for each Ethernet or wireless internet card you have.

In addition, it will define a virtual network interface called the “loopback” or localhost interface. This is used as an interface to connect applications and processes on a single computer to other applications and processes. You can see this referenced as the “lo” interface in many tools.

Many times, administrators configure one interface to service traffic to the internet and another interface for a LAN or private network.

In datacenters with private networking enabled (including DigitalOcean Droplets), your VPS will have two networking interfaces. The “eth0” interface will be configured to handle traffic from the internet, while the “eth1” interface will operate to communicate with a private network.

## Protocols

Networking works by piggybacking a number of different protocols on top of each other. In this way, one piece of data can be transmitted using multiple protocols encapsulated within one another.

We will start with protocols implemented on the lower networking layers and work our way up to protocols with higher abstraction.

### Medium Access Control

Medium access control is a communications protocol that is used to distinguish specific devices. Each device is supposed to get a unique, hardcoded media access control address (MAC address) when it is manufactured that differentiates it from every other device on the internet.

Addressing hardware by the MAC address allows you to reference a device by a unique value even when the software on top may change the name for that specific device during operation.

MAC addressing is one of the only protocols from the low-level link layer that you are likely to interact with on a regular basis.

### IP

The IP protocol is one of the fundamental protocols that allow the internet to work. IP addresses are unique on each network and they allow machines to address each other across a network. It is implemented on the internet layer in the TCP/IP model.

Networks can be linked together, but traffic must be routed when crossing network boundaries. This protocol assumes an unreliable network and multiple paths to the same destination that it can dynamically change between.

There are a number of different implementations of the protocol. The most common implementation today is IPv4 addresses, which follow the pattern `123.123.123.123`, although IPv6 addresses, which follows the pattern `2001:0db8:0000:0000:0000:ff00:0042:8329`, are growing in popularity due to the limited number of available IPv4 addresses.

### ICMP

ICMP stands for internet control message protocol. It is used to send messages between devices to indicate their availability or error conditions. These packets are used in a variety of network diagnostic tools, such as `ping` and `traceroute`.

Usually ICMP packets are transmitted when a different kind of packet encounters a problem. They are used as a feedback mechanism for network communications.

### TCP

TCP stands for transmission control protocol. It is implemented in the transport layer of the TCP/IP model and is used to establish reliable connections.

TCP is one of the protocols that encapsulates data into packets. It then transfers these to the remote end of the connection using the methods available on the lower layers. On the other end, it can check for errors, request certain pieces to be resent, and reassemble the information into one logical piece to send to the application layer.

The protocol builds up a connection prior to data transfer using a system called a three-way handshake. This is a way for the two ends of the communication to acknowledge the request and agree upon a method of ensuring data reliability.

After the data has been sent, the connection is torn down using a similar four-way handshake.

TCP is the protocol of choice for many of the most popular uses for the internet, including WWW, SSH, and email.

### UDP

UDP stands for user datagram protocol. It is a popular companion protocol to TCP and is also implemented in the transport layer.

The fundamental difference between UDP and TCP is that UDP offers unreliable data transfer. It does not verify that data has been received on the other end of the connection. This might sound like a bad thing, and for many purposes, it is. However, it is also extremely important for some functions.

Because it is not required to wait for confirmation that the data was received and forced to resend data, UDP is much faster than TCP. It does not establish a connection with the remote host, it just sends data without confirmation.

Because it is a straightforward transaction, it is useful for communications like querying for network resources. It also doesn’t maintain a state, which makes it great for transmitting data from one machine to many real-time clients. This makes it ideal for VOIP, games, and other applications that cannot afford delays.

### HTTP

HTTP stands for hypertext transfer protocol. It is a protocol defined in the application layer that forms the basis for communication on the web.

HTTP defines a number of verbs that tell the remote system what you are requesting. For instance, GET, POST, and DELETE all interact with the requested data in a different way. To see an example of the different HTTP requests in action, refer to How To Define Routes and HTTP Request Methods in Express.

### DNS

DNS stands for domain name system. It is an application layer protocol used to provide a human-friendly naming mechanism for internet resources. It is what ties a domain name to an IP address and allows you to access sites by name in your browser.

### SSH

SSH stands for secure shell. It is an encrypted protocol implemented in the application layer that can be used to communicate with a remote server in a secure way. Many additional technologies are built around this protocol because of its end-to-end encryption and ubiquity.

There are many other protocols that we haven't covered that are equally important. However, this should give you a good overview of some of the fundamental technologies that make the internet and networking possible.

## Conclusion

At this point, you should be familiar with some networking terminology and be able to understand how different components are able to communicate with each other. This should assist you in understanding other articles and the documentation of your system.