Overlay Networks/ Virtual Networks:

An overlay network is a computer network that is built on top of another network. Most of the overlay networks run on the top of Internet which itself was built as an overlay upon the telephone network, while today (through the advent of VoIP), the telephone network is increasingly turning into an overlay network built on top of the Internet in applications like Skype.

In an overlay network, each of the nodes are connected by virtual or logical links, each of which corresponds to a path, perhaps through many physical links, in the underlying network. Cloud provider networks, content delivery networks (CDNs) peer-to-peer (P2P) networks, Voice over IP (VoIP) services are some examples of overlay networks.

Network overlays dramatically increase the number of virtual subnets that can be created on a physical network. This also in turn, support multi-tenancy and virtual features such as VM mobility, and can speed up the configuration of new and existing services.

Uses of Overlay (Virtual) Networks:

Overlay networks are used in telecommunication due to the availability of optical fibers and digital circuit switching equipments. Overlay networks are pretty much tangled, combining several logical layers operated and built by different entities be it businesses, universities or government. However, It still allows separation of concerns permitting a broad set up of services by different telecom operators/ISPs.

Most of the Overlay networks today are implemented over the Internet. Several protocols are designed for routing of messages across the network without knowing the IP address of the destination. These overlay networks also guarantees high-quality streaming of media through Quality of Service. Also, these networks can be implemented on the end hosts (as required by end-to-end principle) running the overlay protocol software. This network doesn't have a control over the routing of packets between the overlay nodes but it can control the path of the message from source to destination.

RTT of responses from Server:

In a network, in a client-server model, request handling is usually done in the following steps:
1. The Client sends the request (query/command) to be executed to the server through the created socket.
2. The server executed the command and sends the response back to the client.

The Client and Server are connected by a networking link. This link can either be very fast (a loopback interface) or very slow (a connection established over Internet with hops between the hosts). Depending on whatever the network latency is, it takes time for the packets to travel from client to server and back from server to client with the reply. This is called the RTT (Round trip time). This affects the performances when the client needs needs to perform many requests in a row. Consider, if the RTT is “n” ms, due to a slow link, even if server can process requests at a high speed say “m” requests/s, max of m/n. In this case, execution of the same command a number of times on the server.

Also, in an overlay network, the nodes at situated at distant places with a measurable physical distance connected by virtual or logical links. Depending on the speed of these links, connections, the RTT varies accordingly.

Consider the following cases tested with different nodes:

1. When Client and Server connection is established at different location nodes (UDP):
   Server at node → planetlab2.cs.purdue.edu
   Client at node → chronos.disy.inf.uni-konstanz.de

   RTT for client = 1927 ms

   Execution commands used:
   Port number → 13000

command on client – ls /tmp
execution count on client – 3
delay on client – 2

2. When Client and Server connection is established at the nodes within US (TCP):
Server at node → planetlab1.cs.ucla.edu
Client at node → planetlab2.cs.purdue.edu

RTT for client = 230 ms

Execution commands used:
Port number → 14000

command on client – ls -a
execution count on client – 3
delay on client – 5

3. When Client and Server connection is established at different location node (UDP):
Server at node → planetlab-n1.wand.net.nz
Client at node → planetlab2.pop-mg.rnp.br

RTT for client = 447 ms

Execution commands used:
Port number → 17000

command on client – ls
execution count on client – 4

delay on client – 3