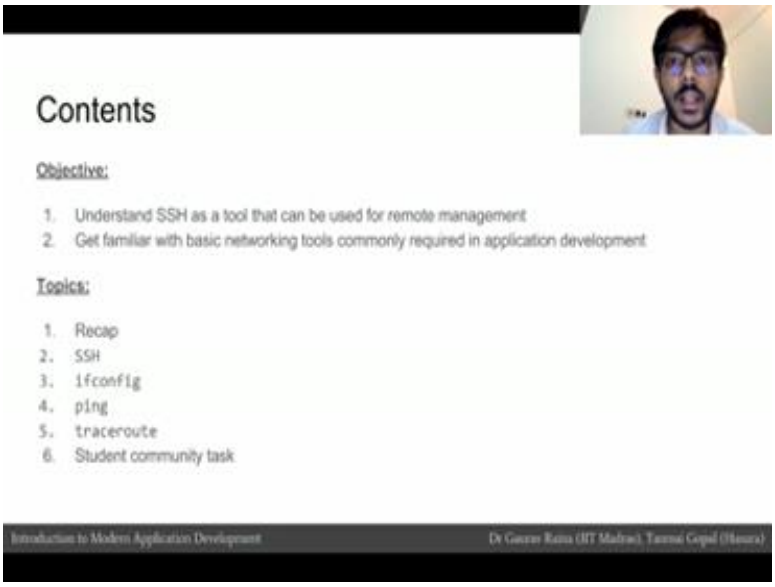
Introduction to Modern Application Development
Prof. Tanmai Gopal
Department of Computer Science and Engineering
Indian Institute of Technology, Madras

Module - P2
Lecture - 08
Practical: SSH + Network experiments

Hi all. Welcome to Module- P2, we will be discussing how to use SSH and going over some network experiments in this practical module.

(Refer Slide Time: 00:09)



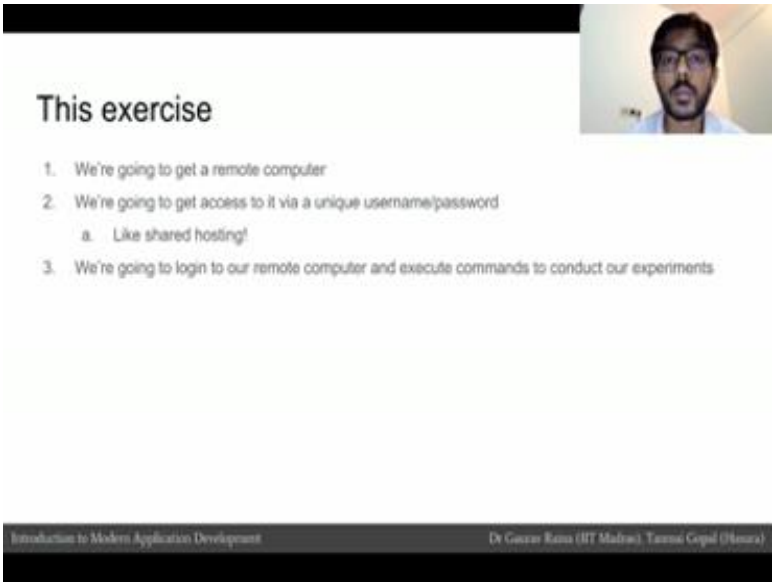
Our main objective in this module is to understand how SSH is a tool that can be used for remote management, and to get familiar with basic networking tools that are commonly required in application development.

(Refer Slide Time: 00:19)

Just a quick recap of what all we covered so far. We have understood that every computer has an IP network, which is used by it to join an IP network. We know that the message is broken down into packets and our center process network and reassembled when they are sent from computer to computer.

We know that these messages pass from a computer to the routers or switches in the network all the way to the destination address and then back. We have also learned that a command line interface is used for executing commands in a computer as an alternative interface to a goi or to a mouse in a keyboard. We also learned that to deploy our code in this server and to remotely manage a server, we need some way of remotely accessing the server and executing commands on it.

(Refer Slide Time: 01:05)



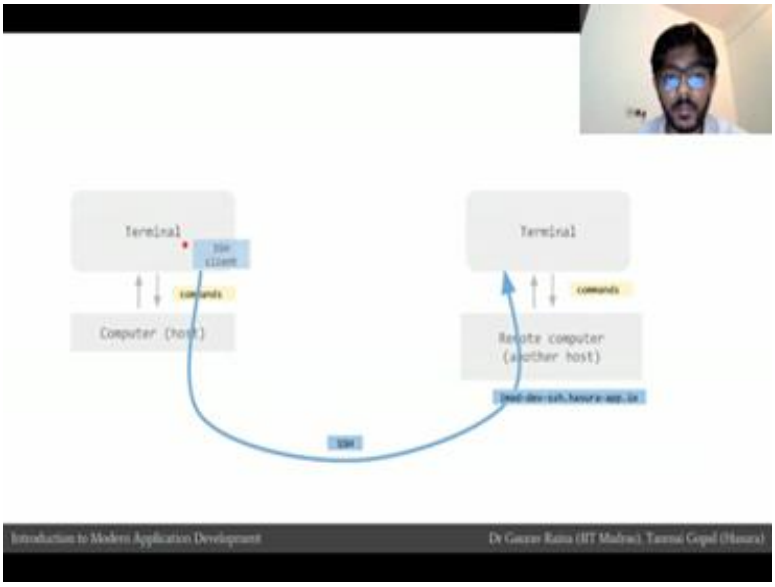
In this exercise, we are going to get a remote computer and we are going to access it via unique username and password that each of us will have, this is sort of like shared hosting.

We are then going to login to our remote computer and then execute commands to conduct our experiments. The first bit which is connecting to a remote computer is important to learn because it tells us of a common method and industry standard method of managing a remote server, the other part of the module is to get familiar with basic networking utilities.

(Refer Slide Time: 01:31)

Let us understand what SSH tells. In the last module we used a terminal or a command prompt on our own computer and we use this to execute commands. What we going to do with SSH? Is that we are going to access the shell or the terminal on a remote computer by logging into that remote computer from our computer, and then execute commands on that terminal on the remote computer. SSH is natively supported on Linux and Mac systems and is one of the best ways of server management.

(Refer Slide Time: 01:57)



Just to make this clear, on the left we have our computer which is also a host in the network and we had a terminal application. We use the terminal application to execute commands on a computer and to get output of those commands and treat them the other terminal.

Similarly we have another remote computer somewhere which is the computer that we want to manage or execute commands on which is also running a terminal, and this terminal application can also execute commands of this computer and read that output. This remote computer has the host name imad-dev-ssh.hasura-app.io and is going to be remotely accessible. This is the computer that all of us will login to and execute our commands.

But how does our terminal connect to this terminal? This is where SSH comes in, SSH is a utility that helps our terminal execute connect to another terminal, and then execute commands on the other terminal directly. If you think about how one terminal is communicating with another terminal over the network, it is actually just the clients, our architecture all over again.

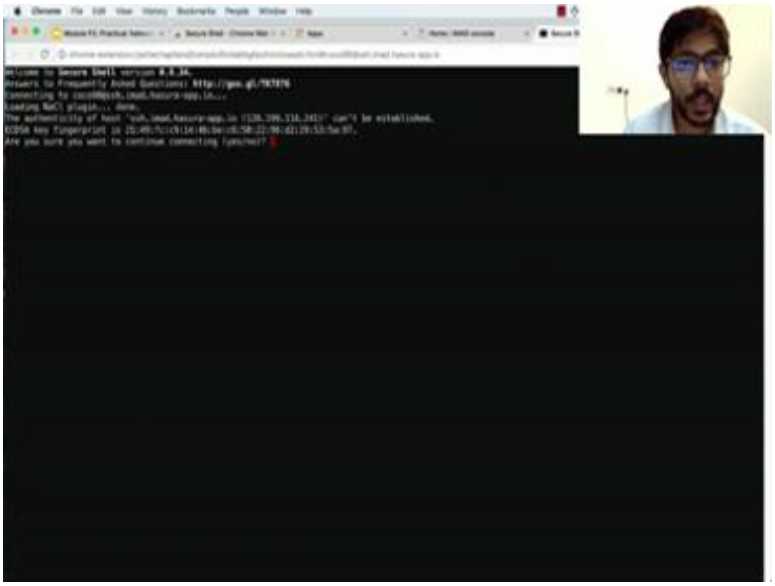
Our terminal has SSH client our SSH client, communicates with an SSH server that is sitting on the terminal. The SSH server interprets our commands that we pass through our SSH client, and passes them on to the terminal on the remote computer, and then executes this commands on to the remote computer. The output of those commands are again picked up by the SSH server, and then sent back to our SSH client our SSH client then displace it on our terminal.

(Refer Slide Time: 03:29)

To access our SSH terminal we going to have to generate our user credentials, which is a username and password to access the remote computer. We have created a console for all the students of this course at cloud.imad.hasura.io. If you login to that console you will be able to create your SSH credentials this console will also be the console on which we will do our application development and our data base exercises, the console requires you to have an account with GitHub. So, please go to GitHub.com and create an account the reason why a GitHub account is required is because when we will be doing the application development portion of this course, the source code will be hosted on GitHub.com.

We will go into more detail about how this will work later you will also need your NPTEL registered email id, to register on the IMAD console. If you have not registered with NPTEL and I still watching this course or still taking this course, you will not be able to access the console and you will also not be able to generate the SSH credentials. But all of this exercise by this entire exercise and this entire module can be done on your local computer by executing these commands on your own terminal application.

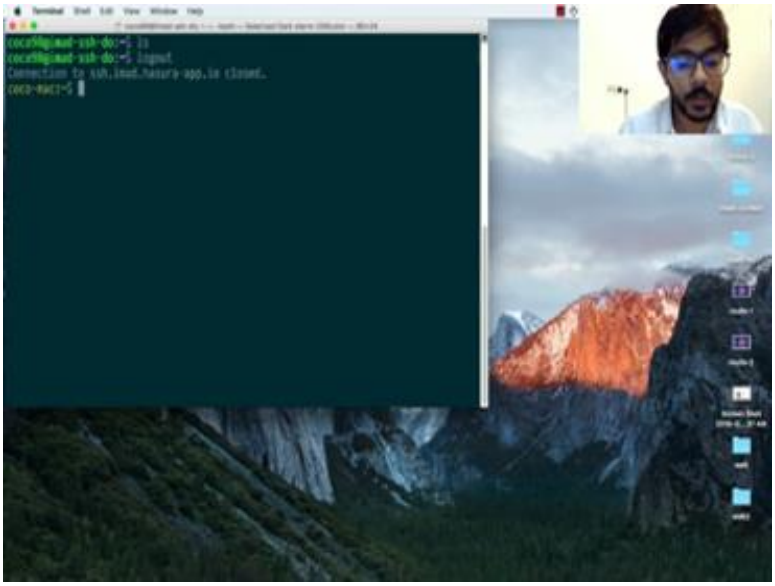
(Refer Slide Time: 04:39)



So, I had to the cloud dot imad.hasura.io link, and I click on sign in with GitHub. I authorize the application for GitHub and confirm the password again for GitHub. Again entered my NPTEL registered email id and I see the console page. Now on this console page later on we will be using the console to create webapp project and to create a database credentials for now let us create our SSH credentials. Now, to access our remote terminal, we have to install a chrome plug in.

So, I say add to chrome this is the secure shell plug in which will create terminal kind of experience inside our browser and allow us to SSH into another terminal, on a remote computer click on add to chrome, here the app now that it is added to my chrome thing I can go back to our homepage. I now click on go to terminal. So, I click on the terminal just to understand what this application is doing, our application this chrome plug in is trying to connect to the host SSH dot imad.hasuraapp.io, and is trying to connect and login with the username coco 98.

(Refer Slide Time: 06:00)



It is asking me if I am sure about the connection I say yes, I then have to enter my password. So, I go and I copy my password from our IMAD console I paste that and in. So, as you can see this is the Ubuntu 16.04 operating system, which is this operating system that is installed on auto mode computer. I can run the same commands that we are running from our previous module and a more typical way of SSH into a remote server is by opening the terminal application.

I am going to zoom that up a little bit, and I am running the SSH command from our terminal which is SSH coco 98 which is my username, at ssh.imad.hasuraapp.io. I am again prompted from my password, and I am going to copy the password from here and as you can see I have accessed to the remote shell here which has coco 98 at IMAD hyphen SSH. I can logout by pressing control d and then back on to my own computer.

(Refer Slide Time: 07:22)

There are several ways of accessing the remote shell, on windows the software commonly used is putty, on a mac the terminal can directly SSH into another remote terminal and on Linux as well. Now let us continue with our networking experiments, the first command that we are going to look at is ifconfig, I had back to my server terminal I type in clear and enter for clearing my screen, and I type the command ifconfig. As you can see two paragraphs have appeared now in the first paragraph says it zero and this second paragraph says yellow.

These paragraphs correspond to the different network interfaces that I have available on this machine. It zero is commonly used for the Ethernet interface. The Ethernet interface is the interface of the operating system that is able to talk to the network card that connects the Ethernet cable or the LAN cable into your computer. The yellow interface is the loop back interface which we discussed from last time, and has the IP address 127001.

So, to analyze this output a little more, we can see that our IP address is mentioned here as inet addr. That is the IP address. Our IP v six address is also mentioned here and we will understand what the IPv6 address is in little bit. Now the ifconfig command is very useful for understanding what network were are connected to and what are IP address on that network is, in case we are on a Wi-Fi network you will have even more output.

For example, in this screenshot you can see there are three interfaces e and zero one which is the Ethernet interface the yellow which is a loopback interface and the Wi-Fi and w yellow one which is the Wi-Fi interface. As you can see we do not have any IP address allocated on this interface because there is no Ethernet cable which is been connected to our computer in this particular case; however, we do have a Wi-Fi address, which is 192 168 1 dot 2. Because we are connected we have the Wi-Fi interface on to our network. This command can be run on your local machines as well, and on windows the command is called ipconfig on Mac and Linux it is ifconfig and on some Linux machines the command will be IP space addr, IP addr.

Just to recap from the previous module what is an IP or more accurately an IPv4 address? The IP is just a 32 bit number a 32 bit number means that we can have a maximum of 2 raise to 32 unique numbers and this means that we can have only 2 raise 23, 2 computers on a single network. And since the internet is just one large network, this means that we restricted to 2 raise to 32 static IP's. 2 raise to 32 is a very large number, but the number is actually less than number of humans on the earth. This means that if each of us wants our own static IP.

We cannot get our own static IP and so, the new standard is IPv6 which is gradually being adopted by everyone across the world. The IPv6 address is a 128 bit address and that is the address that you can see in this screenshot. A 128 bit number has 2 raise to 128 possible unique numbers that can be represented, and this is such a large number that we can assign a unique IP address to each atom on the earth. In fact, not just one earth, but we can assign an IP address for each atom on 100 earths.

So, we would not run out of IPv6 addresses anytime soon. Let us move on to the next command the next command is the ping command which I am sure all of you have heard of in some context of other. The word ping comes from sonar and ultra sound days when you ping an object to see how far away the object is, let us head to our server and clear the screen let us ping www.google.com. I wait for a little bit and I press control c to stop the output.

Let us look at the output of what ping www.google.com gives us, the first thing that ping says is that it is pinging www.google.com which it has resolved for the IP 216 581 964 with 56 bytes of data. It then says that 64 bytes of data were received from this particular

host name which was representing google.com and this response was received by us in 11 milliseconds.

Another ping request was made and a packet was received which was again 64 bytes in 10.8 milliseconds, a total of 5 request was sent if we look at, the overall statistics after we exited the ping program and you can see the control c here after we exited the ping program a total of 5 packets were transmitted and 5 were received which is where we have 5 output lines here, and this translates to a zero percent packet loss. The overall time taken for these request to complete the minimum time was 10.7 milliseconds and the average was 10.86 milliseconds, this time taken for the ping request to complete is called the RTT which is the round trip time.

This is a very important number because it gives us a sense of the latency if the network has between our client and the server google.com. So, our client which is this machine that we have SSH into is contacting the server google.com, and for the packet to go all the way to google.com and come back the round trip time is about 10 to 11 milliseconds. To understand RTT a little more the time that is theoretically taken between a client and a server over a network is going to be the speed of light or the speed of electromagnetic waves, which is electricity in the medium between our client and server.

Now if this is copper or fiber the speed of light is a fraction is a fairly large fraction of the original speed of light. So, about 0.9 times the speed of light and so; that means, that the total time taken for request to go to the server and come back from our client is at least and a minimum of twice the distance between the client and the server twice, the distance because the distance to the destination and back from the destination divided by the speed of light.

This would give us the minimum amount of time taken for a request to for a request to reach the server and a response to be received from the server, and this means that from Chennai to San Francisco the round trip time has to be a minimum of hundred milliseconds because that is what the speed of light tells us, and it is not possible for a response to be (Refer Time:13:57) in that which means that our site if the site is being served only by a server in San Francisco the site cannot load faster than a 100 milliseconds.

Next let us look at the Traceroute command, the Traceroute command is not a command that we used very often in application development, but it is a very interesting command that tells us a lot about the network. Let us head back to our server clear the screen; I am going to run a traceroute on google.com first.

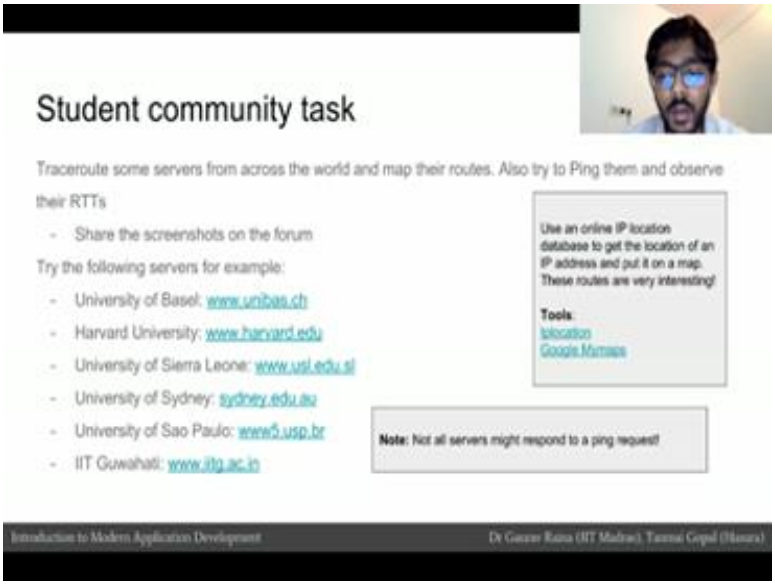
Let us wait for the command to complete, let us read the output of the command again. So, it says that in the traceroute to www.google.com, which was at this IP 21658 196 dot 196 wends through 8 particular hops, it first reach the IP 128 199 127 253, which is in the data center, where this machine is and then it wends through several routers or switches till it finally, reached our destination. If you run the traceroute command locally on your computer this is the kind of output that you will see.

So, in this screenshot we have done a traceroute on www.gmail.com and the packets are making their way to the destination which is here, the first place that the packet reaches is 192 168 1 dot 1, which is the IP of a router. So, from our computer the packet reaches a router, from there on it goes on to the ISP and this is and since we using the internet provider a c t it is going to the ISP's routers and from there on it is making it is ways to some other routers over the internet till it finally, reaches the Gmail server. You can run these commands locally on your machine on a windows machine this command would be tracert and on mac and Linux you can reduce a traceroute.

(Refer Slide Time: 15:59)

I run a traceroute on ug.edu.gh which is the server for university of Ghana and I took the locations just to go back to the previous screenshot. I took the IP's in the middle and I plotted those locations on to a map, I converted the IP's two locations using an online database that tells us the rough locations of every IP, and after putting it on a map we can see that this is the route that our connection took to go from Bangalore to Ghana and we can see if the connection run from Bangalore to Bombay to Marseille in France to Paris to London and then from London to Ghana.

(Refer Slide Time: 16:38)



I would request all of you to and to ping and traceroute the servers that has been mentioned on this slide, and to post the RTT's and to post your observations about the routes on the forum. If you do have the time please try to plot out some of these routes on to the maps and share the map locations you will discover a lot of new and interesting things about the way the internet works after going through this exercise.

To convert your IP's into locations, you can use an IP location database there are several of them available on the internet the one that I used was IP location dot net, which you can find by clicking on the link on this slide. I then used Google my maps which is again accessible from the link on this slide to plot out the route on a map. It is important to remember that are not all servers might respond to a ping request this happens because those servers are configured not to respond to ping request, and do not have a ping server running or blocked the ICMP protocol.

(Refer Slide Time: 17:41)

To quickly summarize in this module we learnt how to access a remote computer and execute commands on the remote computer via SSH. We also used the ifconfig tool to understand the host network configuration, the ping tool to ping the remote server and get the RTT, and we also used the traceroute tool to see the route that our packets take through the network to reach the destination.