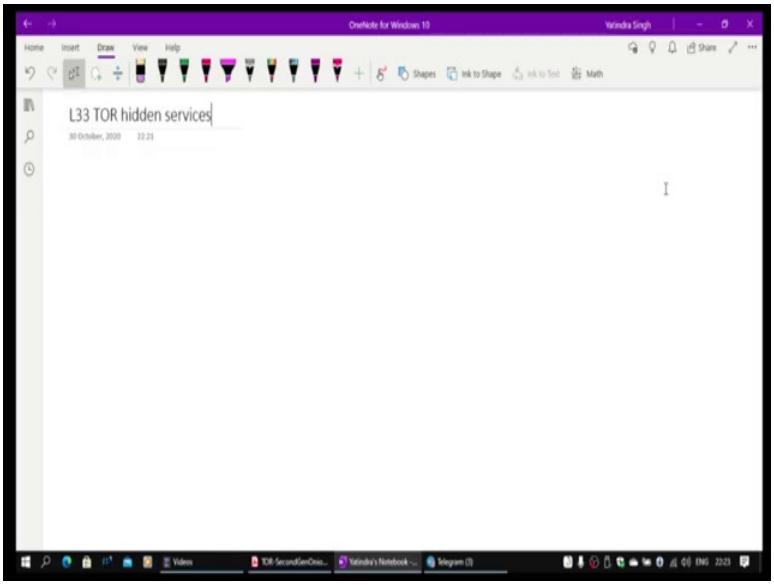
Peer to Peer Networks
Professor Y. N. Singh
Department of Electrical Engineering
Indian Institute of Technology, Kanpur
Lecture 33
Hidden Services on TOR Network

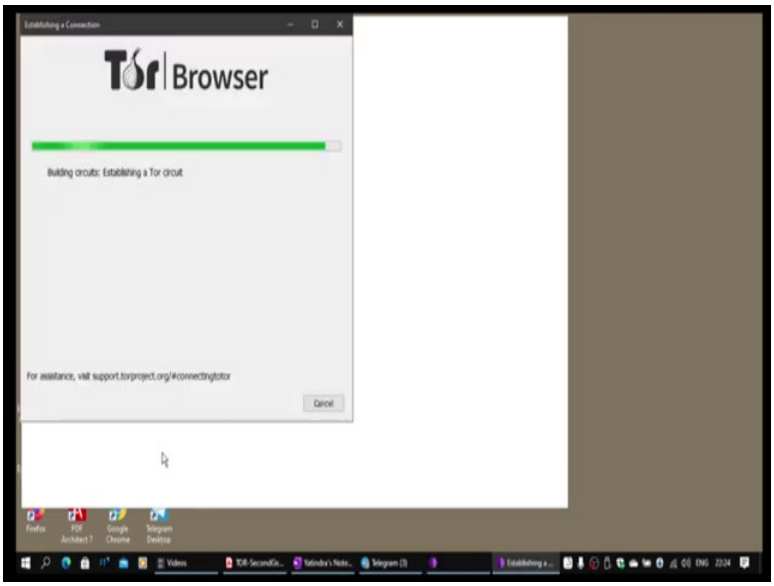
Welcome to the second last lecture of this MOOC series, where I will be technical part, I will be closing here in this one, and then I will give a summary lecture, kind of giving an overview of whatever we have done in the whole MOOC. This is a kind of putting everything in perspective about the technology that I have introduced in this MOOC. Probably, in future versions of the MOOC series, I will have even more refined and more structured version. So, this has been given was given for the first time so it was also a good experience for me. A learning experience, I should say. And ofcourse, some of the feedback has helped me in improving.

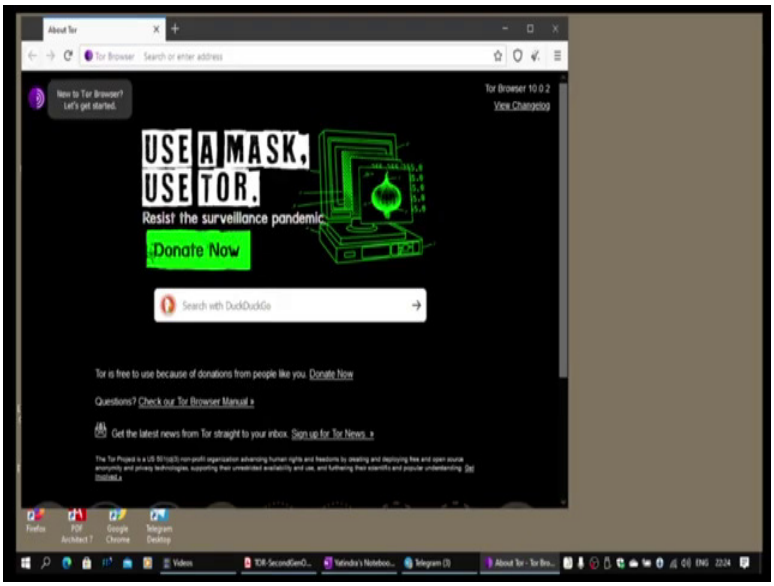
(Refer Slide Time: 00:56)

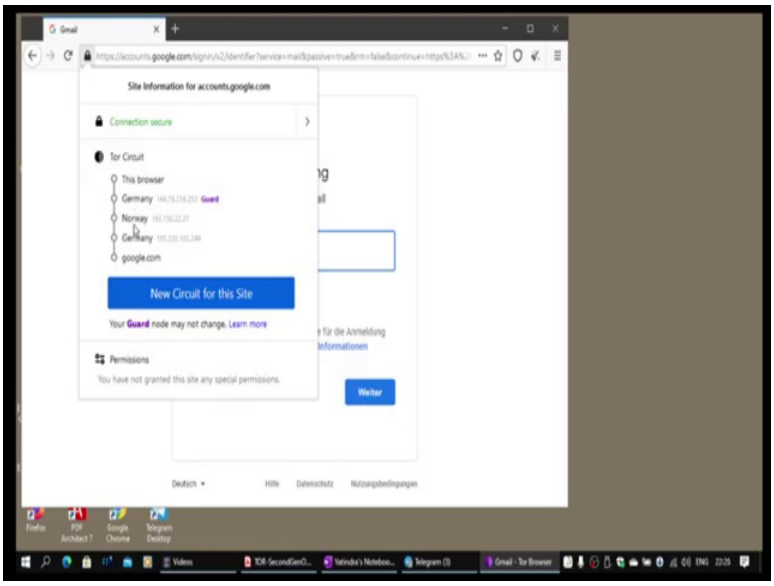


Now we will be talking about hidden services. So, basically lot of us have already heard about dark net. So, what is that essentially it is all about that thing? Before moving ahead with how these hidden services or Darknet websites are being created and they actually operate how the protocol operates. I am not telling you how to create it, but I can certainly show you how these operate.

(Refer Slide Time: 01:19)







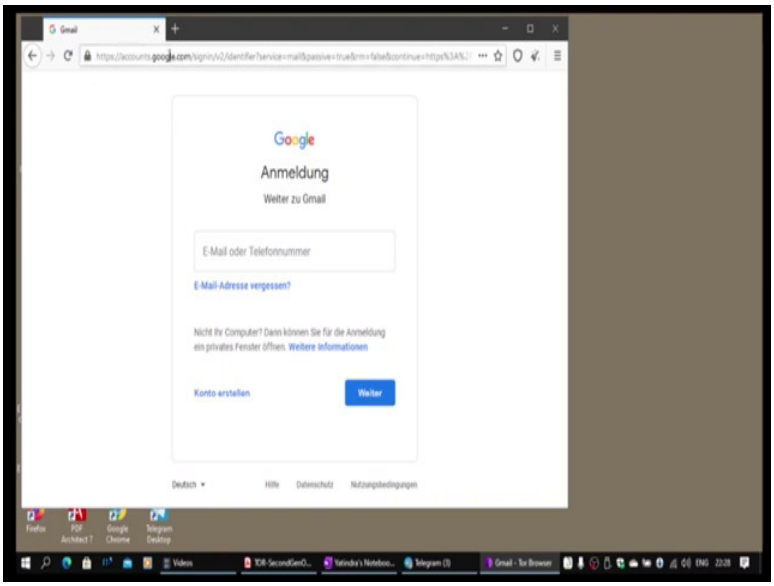
For this you need a tor browser, so I have installed a tor browser here, it actually comes even on an Android phone. Once it is basically Firefox, it is not installed in the OS. It is installed in the user space. Once I click, it is going to start. And normally, window size we should not touch if we want to keep anonymity and there are many issues which I have not discussed, there many questions. Now as you have seen, that it was building up the circuit this how it will look like. This is the latest version which I have as of now.

If I want to browse a website, then it will be done anonymously. Let me for example, I want to check my Google mail. I am actually doing it from India, but I do not know from which country the exact node will be sitting in which country. The Google actually the language shown will be different depending on that. And Google actually will immediately block my account if I try to access it through this, because I am currently India. I am not supposed to immediately access the same account from some other country.

They do keep a check so the test can be done. I am not doing it. It is now searching is actually forming a circuit as of now. Now it has already created a circuit. And I can click on this and then get the circuit. There is something called guard. This is for some protection against then attack. So, this guard roughly will remain for 9 months, will be remain fixed for me, the remaining circuit will be keep on changing actually. I have not discussed about this guard this was something new.

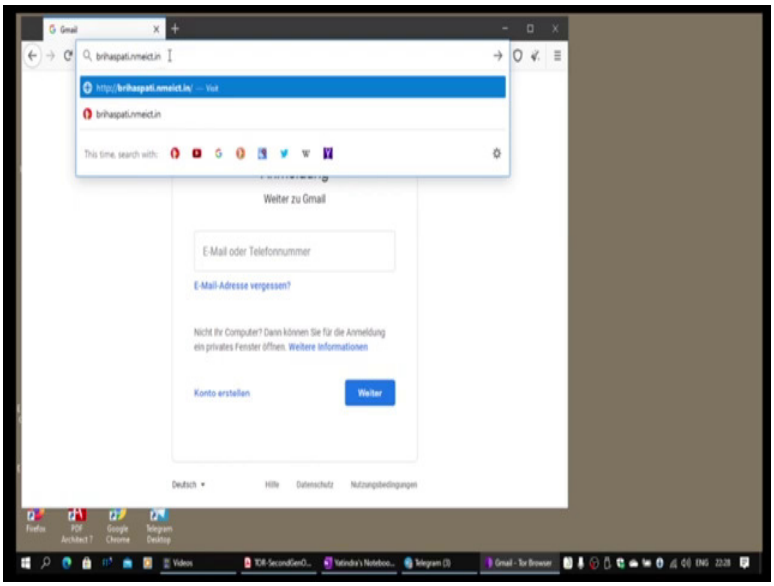
Now, this ultimately because from my browser in India, it is going to Germany, some other TOR relay from there it is going to Norway. But these guy do not know that, the Germany guy knows that it is coming from my IP address, the guard. But Norway person will only know it is coming from German, this particular IP address, and then from Norway it is going to again back to Germany and then from there it is going to Google.

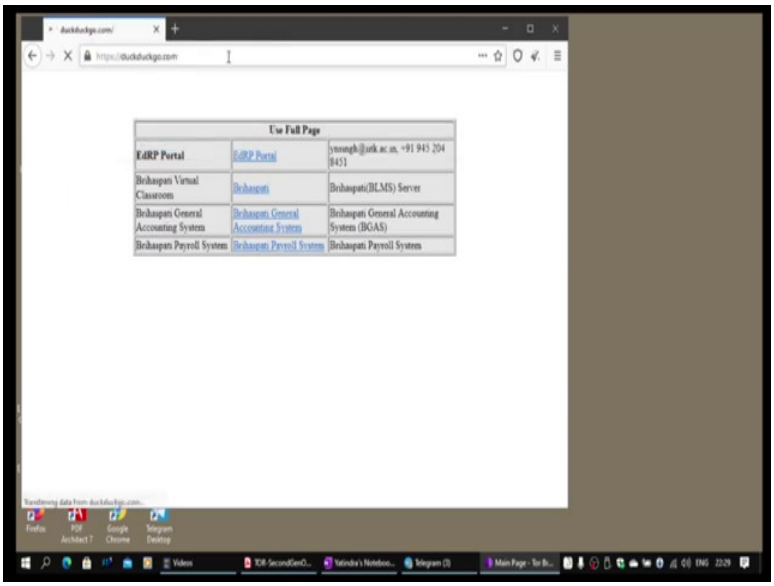
(Refer Slide Time: 03:32)



So, now you can see because the Google account is now coming German because of that. And there are even essentially so now this is one particular specific website. So, this was actually browsing the Internet anonymously. So, the Google will not know that I am actually browsing from India.

(Refer Slide Time: 03:51)



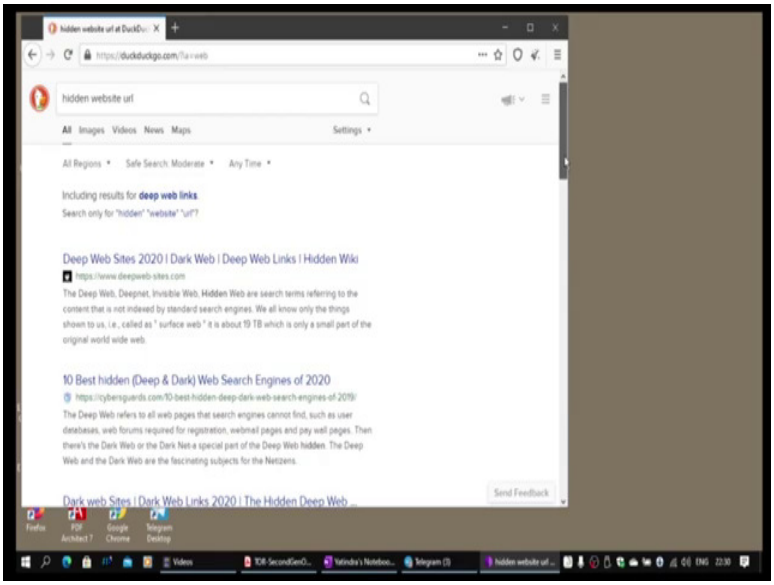


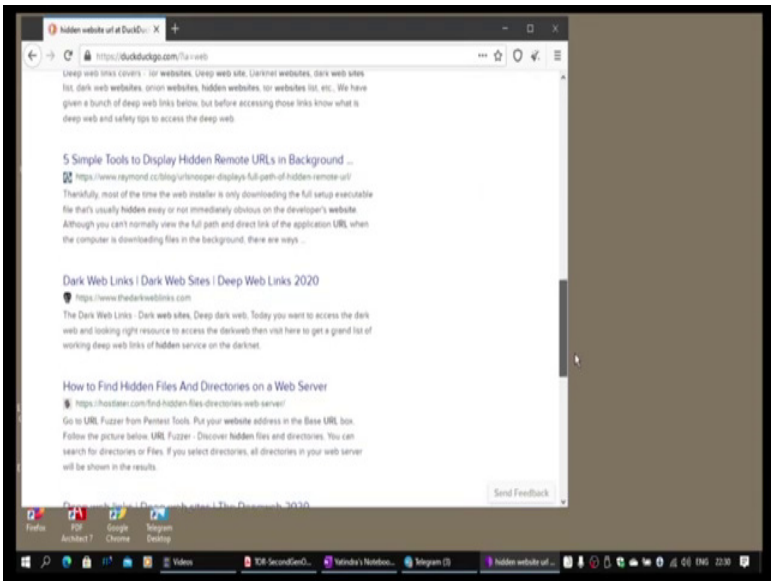
So, even for example, this is a system which we have done the Brihaspati thing. Let me try to access it, it is within India but the access because the guard node normally does not get changed. The relay will be done through that so it is unsecure system. But again, it is going to guard Finland, Belgium and from that Belgium to anonymity has been accessed. It is where is our system and connection is not secure because I have not done https. The Belgium proxy can actually identify that what is being transacted with Brihaspati. Normally if I am not doing that, I should always use https not http.

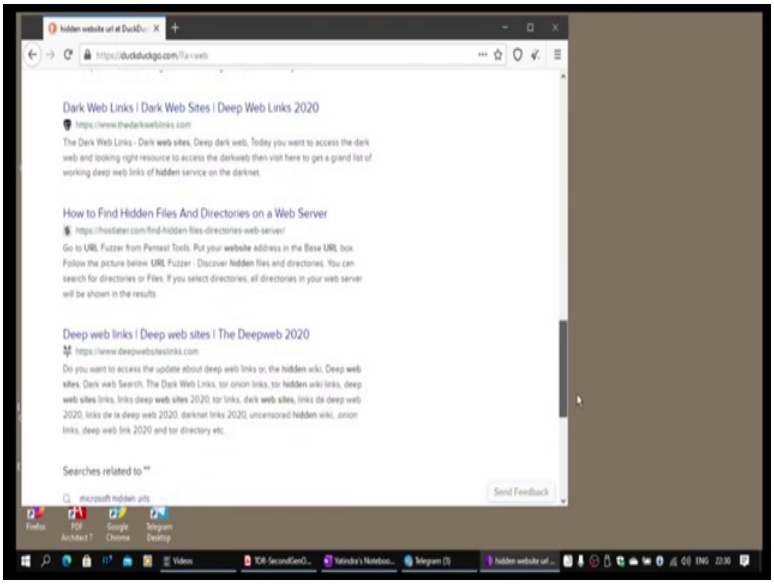
That even in the last exit node should not know what communication is happening. So, because then it become end to end encrypted. Now this end to end encrypted and actually it is still going through the same place. Now, there is another kind of websites which are called hidden websites. And that is what I am going to talk about today that how those websites operate. So, there are actually a few of them. Normally, these websites, URLs are not publicly available, but we can try searching.

This I am using tor browser it goes by default to duck-duck go search engine not to Google. I can actually put a key string here. I can find out some hidden website URL. And remember when you close this it will remove everything. It do not maintain history, it do not maintain cookies everything gets erased. There is no record actually in this case.

(Refer Slide Time: 05:23)



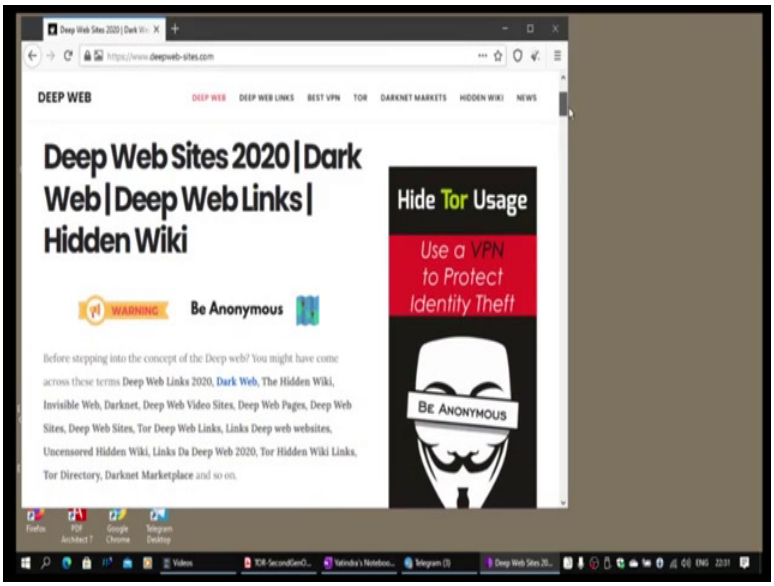


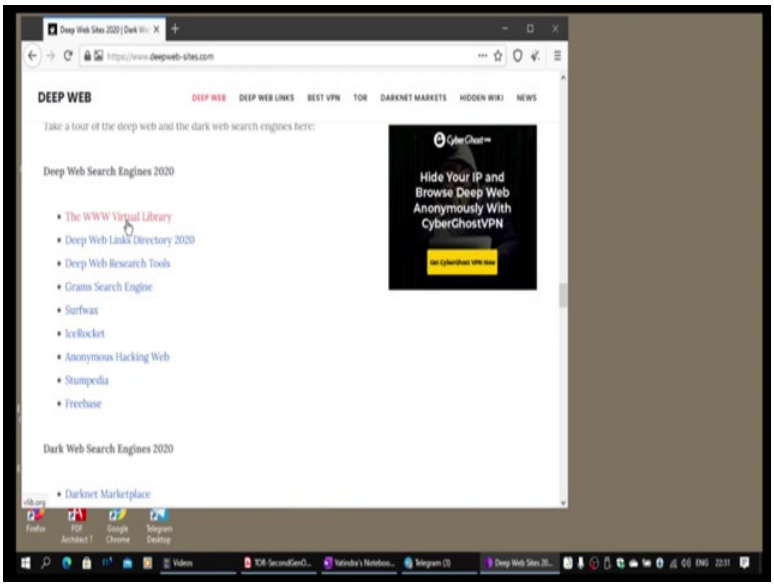


These are kind of websites which are there. I would like to show one example of an Onion website. And this is also called deep web or the dark net. That is what it means, and all URLs of Darknet will be some random string dot onion that is the way the URLs are there. So, normally the browser will look at this dot onion top domain name. This is should actually immediately figure out it is there is no designative resolution to be done in this case.

It has to be done. The way I am going to explain in the lecture how this actually operates. It has to be searched in that fashion. So let us search for some website. Let me see if I can find something meaningful and then we can use it.

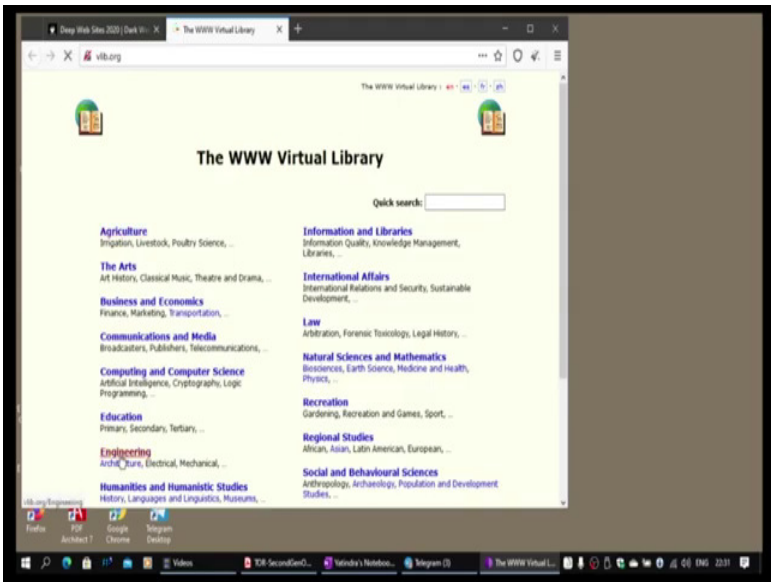
(Refer Slide Time: 06:12)

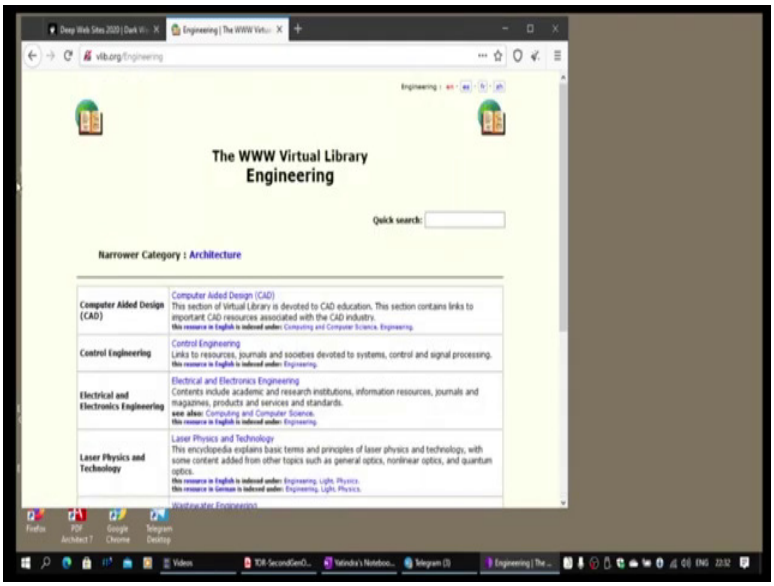




I have got some place there is actually common website, as you can see this is on the Internet. You can also browse it but it actually also gives a warning that these are illegal sites so I will not be doing much of it here for you. It gives all the stuff. There are many kind of thing. You can choose anything which you wish so. I only just look at their virtual library.

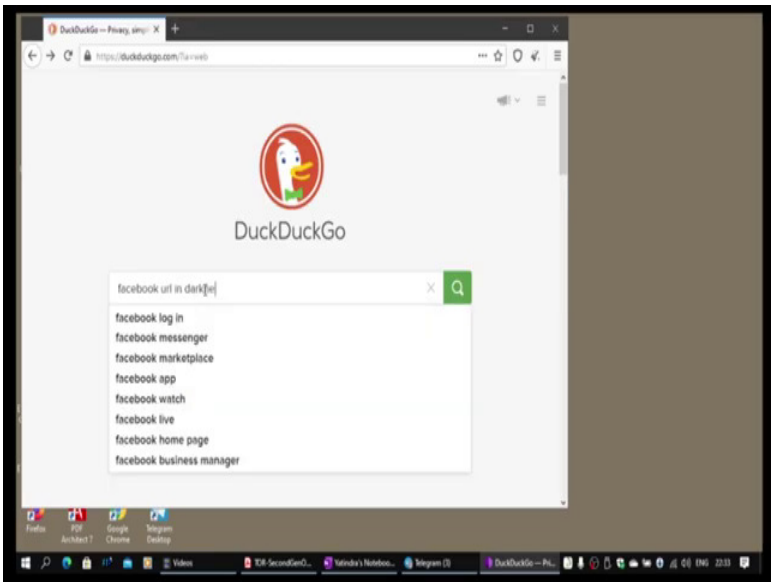
(Refer Slide Time: 06:43)

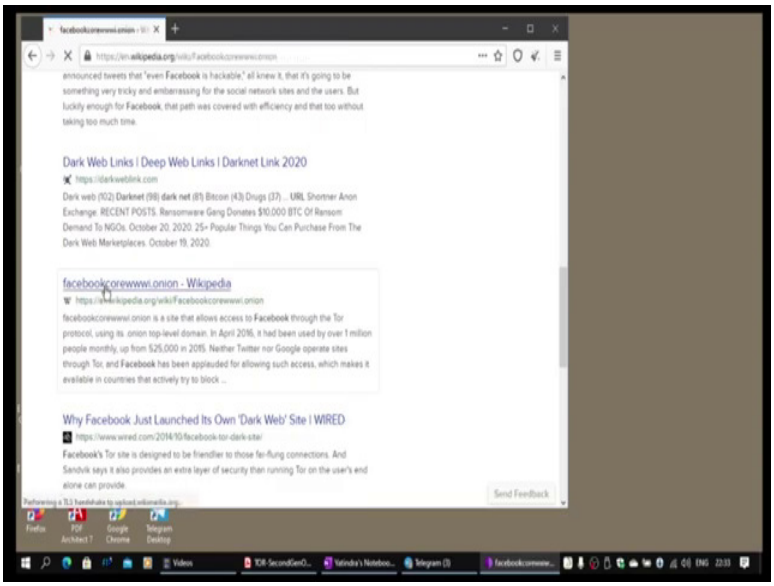


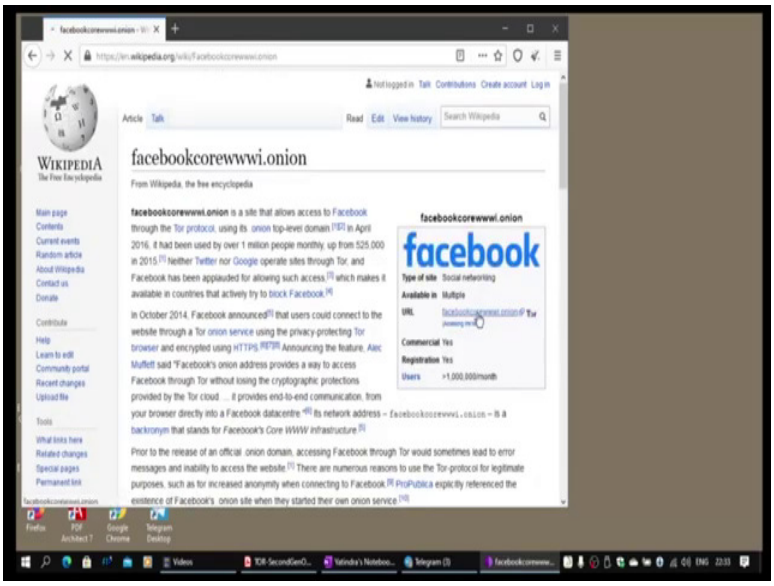


That is the deep web search engine. Let me look at something related to engineering. And, let us see if I can find something. Yes, so now here actually you get some of them. This is a control engineering it says; no it is not a deep web. Let me just look at something else. There are some links which will be available here which can be used. Normally, it is kind of very not that much structured the way it happens regularly.

(Refer Slide Time: 07:28)



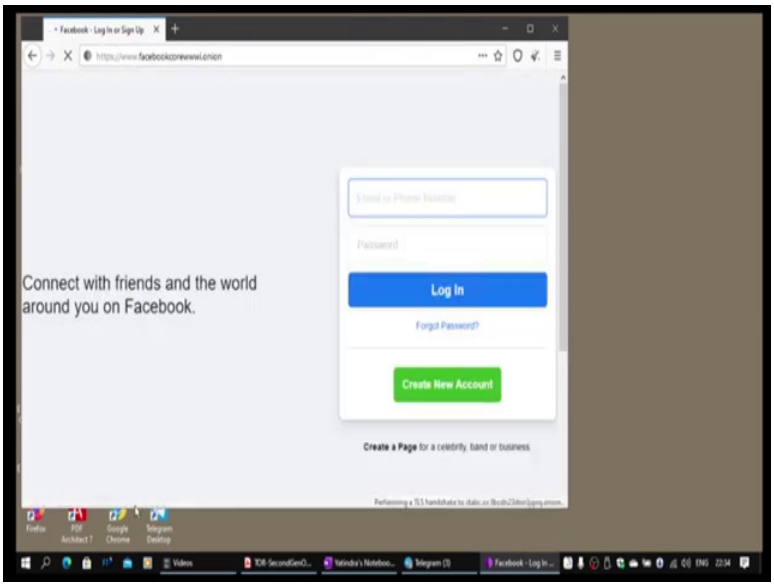


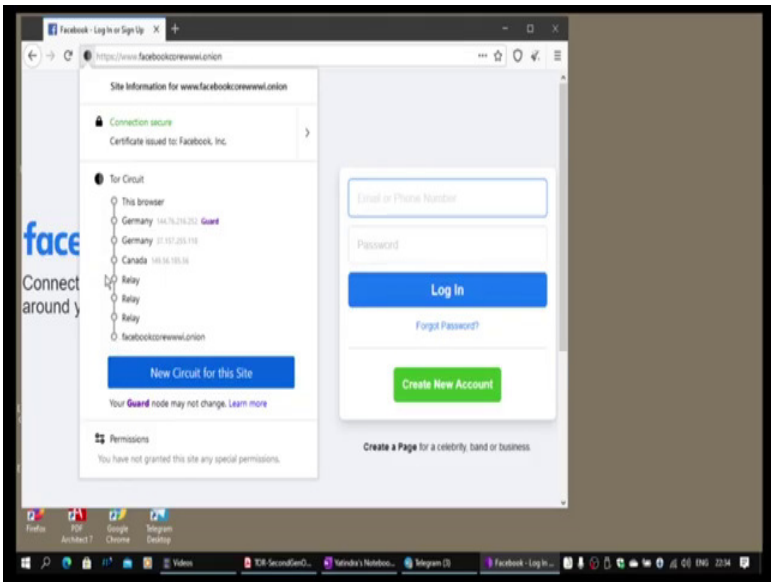


You have to do kind of brute force thing, but one thing is there the Facebook is one organization which has provided URL which can be accessed when you are actually behind. For example if we are in China or North Korea. You cannot access it directly because being blocked. So, you can then actually use a tor website of Facebook and still access the Facebook. So, Facebook actually is providing services in dark net and it has a URL for that. So, you can search for that Facebook URL and try that actually.

Yes, it is actually again talks about this. And should be able to find it; yes, this is the URL, so Wikipedia gives this one, and this, ofcourse one can try. I am now connecting to the Facebook, which is operating in Darknet, it is very interesting, this is a dot onion website. This is not cannot be accessed by any other browser. If I use normal Firefox, I would not be able to access it.

(Refer Slide Time: 08:34)



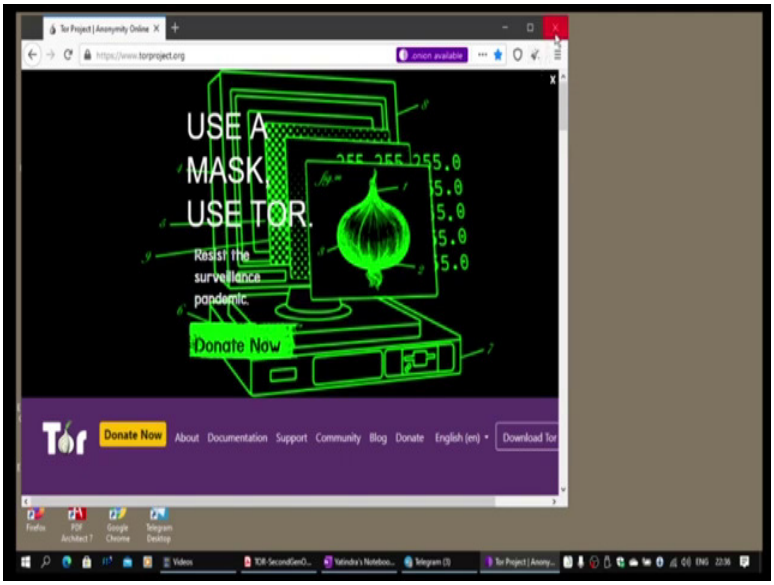


Now, this essentially has been mapped to a different website. This you started from here, it is actually connecting through a Darknet. Now you should also understand these relays will not be available to me. So, there is circuit being created. There is a Facebook thing last end point, this, is the as usual my own guard node. And at some point there is going to be splicing of the two connections which are happening. So, for there is a telescopic encryption channel which is been created by my thing and this one has been created by Facebook, the hidden service, server, and they are being spliced together.

This is essentially is the key behind how the things work. And now it actually has not remembered anything. If I tried it will ask all kind of verification from me before I can log into the system. Now this is a hidden thing, there are a lot of things which run. And I think you have to be careful when you are doing it. You cannot relay there is no verification, there is nothing. So, take your own risk if you want to actually go through this.

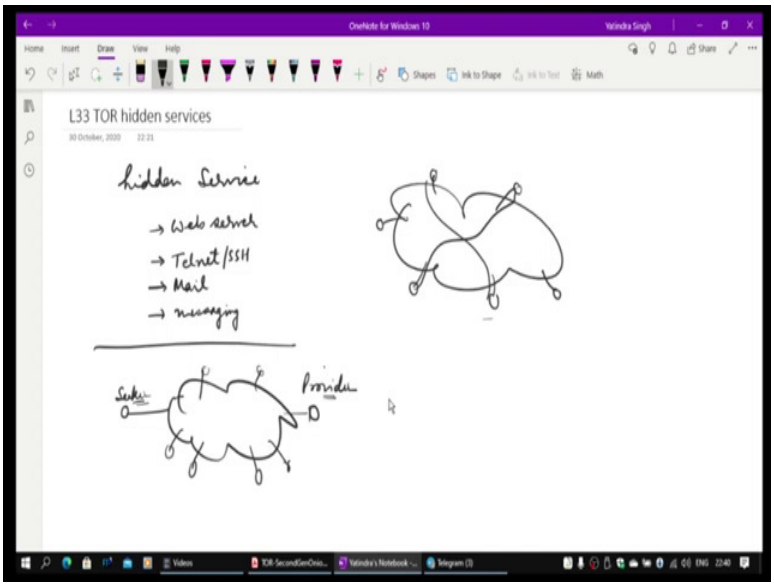
You can set up your own Darknet website if you wish. I am only going to talk about the scientific aspect of it or the engineering aspect of it, which is pretty much interesting, can be used in certain applications. Let us move to the lecture now, this was the demo and this one is the tor browser I get. I actually can as I mentioned, you can always get it from tor project dot org from there you can download.

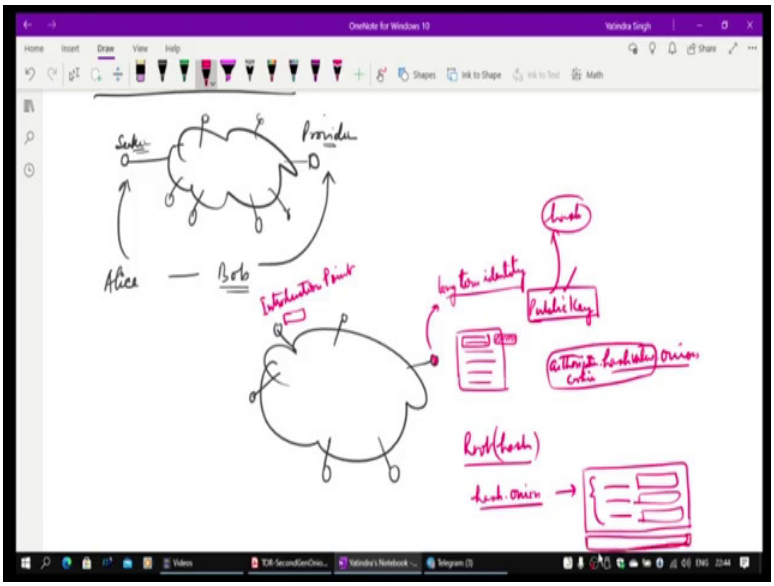
(Refer Slide Time: 10:13)

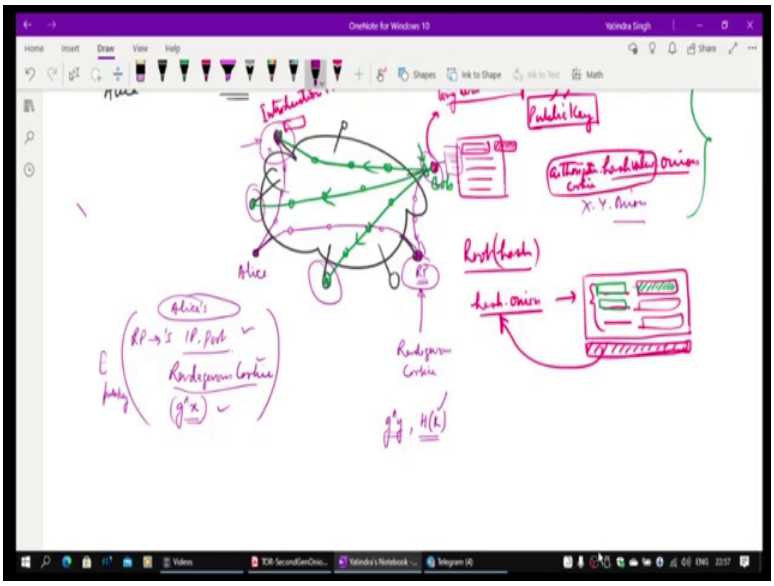


I have done actually from there itself. You can download the browser from here, so anyway let us close it and once I close everything will be lost, everything is, there is no fragments which are there on this machine or whatever I have browse everything is lost, that is anonymity. There is only installation, which is in the user space and Microsoft will not be able to figure out that it has been installed. It is like a separate application running in my directory which I start. It is, having nothing linked to whatever is the data structure of the operating system here.

(Refer Slide Time: 10:49)







Now, let us move to the how it actually works. So, these hidden services are also called onion websites as I have explained, and these onion or hidden services need not be hidden services need not be only web service the way which I have shown.

The normally people will have impressions the Web service, no, you can actually put another kind of services also you can actually do a telnet so you can do a remote computing machine, but it is a hidden machine and you are providing that service. We can have, in addition, a mailing system, you can actually send mails dispatched to a hidden machine, hidden service, you can send any other kind of say messaging; instant messaging can be done.

You can actually have gaming being done. Basically you are creating TCP circuit to hidden services actually. So, you can actually have kind of can also create a network where everybody is hidden. And through hidden, so nobody knows who is where, but you can actually now do a peer to peer connection between them. So, anybody can be server anybody can be client even that kind of peer-to-peer, anonymous peer to peer network also can be built, ofcourse. But only problem is that you have to ensure because there is a anonymity which is there, there is no identity.

Identity can always be a dropped and new identity can be acquired. Even reputation system does not work here. So, far, it internally both people mutually authenticate and then do the work, and ofcourse it is fine. You have to essentially trade very carefully when you work with these kinds

of systems. Now important thing is that how this is there is service seeker which is looking for a service, maybe a browser trying to browse a network.

There is a guy who is providing a service, there is a provider. The important point here is that this guy does not know who the provider is and provider does not know who the seeker, service seeker is actually. And there are some intermediate onion routers through which this service has been provision. So, they both are hidden to each other and nobody in the internet also anybody is not knowing that who is asking service from whom and what he is asking, actually.

That is also not identified being him, there is a pure anonymity. This is what essentially is being now done through hidden service. One of the important things which is done is, for example, I will take the same names I am actually using which was has been given in the paper from where it has been studied, actually. And as things are also changing continuously so next time, if I teach the same thing, lot of things would have got modified. It is a very dynamic thing.

There is a Alice and Bob they want to communicate. So, Bob is provider and Alice is the service seeker. And we are taking it like a website or any service, it does not matter. Now Bob has to essentially figure out in the network. Bob will sitting somewhere. There will be lot of other onion routers which are represent. So how people will connect to Bob? So, that, the way it has to be done is, Bob will essentially will now generate something called introduction point.

Somebody wants to have the service has to essentially go and connect to the introduction point. This Bob has to decide while it will actually get the list of nodes from the directory server and that list of nodes it will pick up. These guys are going to be my introduction servers. And then it will also have a long term identity. This long term identity is; the way we have used earlier it was actually node ID.

But there is no IP address need not be told to anybody, actually. This is a node ID. And it can use, in fact, a different another public private key pair and that public key can be used to identify. Public key can be used to identify the service, basically and the way it is going to happen is normally, these introduction points list, and for each introduction point it can have a separate public key, public-private key pair which can be listed here for the same service.

And where this is going to be stored? So this public key is hash can be computed. And this hash value dot onion that can be the URL in fact there will be also an authorization cookie. So, that is what has been mentioned in the paper, but I have not seen any authorization cookie anywhere. Normally this would have been a string dot onion. I will assume that this is one common thing we in fact seen just now for the Facebook that is the way it was.

This hash value is where this has to be index so there is, DHT index which can be maintained by nodes. This guy can actually publish this. It will find out the root of this hash value root node of this. This can even go in the directory servers actually. For this hash value will become the key. This will maintain or this hash value dot onion it will maintain these are the service providers.

And these are the corresponding public keys which have to be used, the corresponding private keys are already available here and this will be published by. This will be digitally signed by something. Somebody so that it is authenticated, even if it is not done ultimately you can always. There is a public key with which you can actually verify this key and public key is hash has to this. Then only this is going to become valid.

And this is going for individual communication which is going to happen from each introduction points there can be many of them. So, it can actually create one here. It can create one here, one here, one here so many introduction points can be created. Now what it will do is it will now set up a telescopic tunnel. So same way it go to first node then actually create and create it will happen and then it will do relay extend it will go to the next guy, extend next guy and ultimately it will extend here, it is a telescopic tunnel encrypted one being created all the way to this various introduction points.

So now anybody wants to access the service has to come to one of these introduction points. So, now this key value pair will be used, can be stored in directory servers or it can be in a DHT index here through which anybody can find out which introduction point I have to contact. These are the options you just contact anybody and this is the public key to be used when communicating. Then of course service identifier which will be available and ofcourse the introduction point.

The service identifier the public key and then ofcourse, the introduction point IP address, port number, everything will be there and this guy will be maintaining telescopic tunnel. This introduction point does not know where the server is actually. There is a onion proxy of this server which will be doing this job. Now the Alice will do what? This is, that is what Bob has done. Let us see what Alice does.

Alice will do a trick here, he will if this is the Alice he will figure out these are introduction point I have to communicate first she will find out some node which will become responsible for as a rendezvous point RP. And she will then create this is Alice. And she will create a telescopic tunnel through some nodes all the way to rendezvous point. And this will submit a rendezvous cookie will be deposited here. The rendezvous cookie will rendezvous point or rendezvous node will allow any request to come in, and if the cookie is being presented than simply splice that connection here.

Bob essentially is now supposed to create telescopic tunnel all the way here. And present the random cookie somehow and then the connection will be made and then you will have a telescopic tunnel all the way to the random point. And so this is encryption, maybe say four layer of encryption, then 3, then 2, then 1 and then is being spliced and then one layer, then two layer then 3 layers then 4 layer and Alice will do all that decryption and there is end to end encryption which will be happening.

This is what is supposed to be done. But how this will be happening? So, Alice will first of all create a telescopic tunnel to safeguard its identity and come here. So, once she comes here and remembers before that this Alice has to figure out from directory servers what service she is trying to look at. Now if she needs to maintain anonymity, this even searching in the directory server on the DHT table she has to maintain.

And so she needs to actually make a telescopic tunnel and ask some proxy to do it on her behalf and get back the results. So, that she will remain anonymous otherwise in first step itself some people will be able to figure that she is trying to search for this particular service. So, remember when I was using tor browser, I was actually trying to find the service, but through tor browser not through my normal browser.

And then using that particular URI there in the tor browser now that is not safe you have to do it from the tor browser itself. So, this is something similar. So, Alice will now create a find out this introduction point. And she will then actually transfer some information to the Bob. So, this guy always is waiting. So, she will actually give information about the she can give information about herself. She need not give actually. So, and once this information is there this information will passed. We will see how what all information is required.

Now, important thing is that now we know as of now that there is going to x dot y dot onion kind of thing the address which will be used. So, which is going to be authorization key? We have already mentioned that. The authorization has value x dot y dot onion that will be the structure. So, now what she is going to tell to this bob? So, Bob's actually public key will be already known from this table, from this directory server or from DHT it will figure out, so she will encrypt the rendezvous point information.

What is RP? Actually IP address and port number that will be informed by her. He will tell about rendezvous cookie. She will also give the first half of the Diffie-Hellman hand shake starting point. So, this will be all encrypted by the public key of which is being provided by the in the case of this data structures, which are whatever service descriptor which has been published. This actually will be encrypted by that and this will post.

Once this actually comes back to Bob knows that it has come from here and my correspondent the public key, which I have mentioned here, the corresponding private key is with him, he will decrypt everything he will actually decrypt with all this tunnelling layers. So, In fact, for me, it is a secured tunnel which provides anonymity. Once it comes back, it can decrypt all this information. And if Alice has an option to give information about herself, if she wants, she may not give actually it is her choice.

And depending on that, Bob may decide whether I should respond or not respond if it wants to respond it will then actually create an anonymous again an encrypted tunnel. This is telescopically encrypted tunnel all the way to RP and once it gets connected, it will then splice the, it will present the rendezvous cookie which is received from here. It will also send the second part of the Diffie-Hellman hand shake and it will also send hash of the key which it has actually generated.

So, even if this g, y can be identified by RP but it cannot find out what is the key, which has been computed. So, this x is only move to Bob as of now, Alice has already sent her this thing encrypted from here itself this encryption was done for the Bob and it was went through this way. Even no node in between actually is knowing this information. This information comes back here Alice will decrypt everything.

It will know this information so it can find out what is k. Bob also knows k. Now whatever information is there it will go all the way it will do relay data actually for this thing. And this is actually is now connecting backs to this another circuit, which is a kind of receiving. This is either acting like OR gate for this as well as OR gate for this so there is a connection splice which is happening. So, this connection splice happens, which is encrypted with this and this will keep on happening ultimately till this point it will decrypt and ultimately it will decrypt with this key. Information is available which can go to a server.

So server will now give the information back which will go to Alice. Now, Alice is not knowing where the bob is and bob does not know where the Alice is or who they are unless they mutually agree to identify each other. But they will only users will be identified but the IP address and port numbers will still not be known unless they find out their IP address and port numbers and tell it via the communication channel. But communication channel itself will never tell it actually. There is no way the protocol wise you can figure it out.

Now, they will start communicating and this how the hidden services actually work in the system. And this is pretty powerful technique and this is very good application. You can anonymously talk to other people and this is what the Darknet. There is lot of website, which are running, but be careful whenever we are using it. And there are still a lot of problems which we have to figure out, we have to solve them. And still, there are many kinds of attacks which are feasible, which needs a resolution.

In fact, for example, we are actually going through these routers there is condition control has to built-in inside the system. Some nodes is having less capacity. So, what has to happen if it is not able to exit so there is a flow control, which is to be built? So, people should not be overwhelmed at any point of time. It should work smoothly. Secondly you are taking through so much along hop so much of encryption, decryption happening, even if whenever you are doing

AES base encryption with counter based method, not at actually even then speed is expected to be fast, but it is not that fast.

Performance is not that good the way it happens when you are doing direct connection to a server. So if you want to, very high bandwidth is required how to get it while maintaining anonymity and without getting into the eyes of somebody who is actually doing surveillance on the network. So, it remains a pretty standard problem with that I think I close this lectures and I have described how the hidden service conceptually actually work.

So, it is technically, the idea is that you always create a random generated rendezvous point and the service seeker connects to that and service provider also connects to that, if you want to provide the service. Now introduction points are not going to be liable because there is no data from Alice to Bob is been transacted through this. Now this guy is, for example, selling drugs that information is never passed through this guy, so this guy is never liable for that or is doing something else, some nasty thing.

This guy is not liable it was just only introduced them, so actually the RP is doing. But RP is hidden it is not a public entity. In fact, nobody is public entity here. But you can always find out this guy because, you know, this person's IP address and port number is available. So, this need to be safeguarded. RPs IP address and port number is only known to Bob and Alice. So, kindly to safeguard these introduction points we use, this particular design is what is implemented in the system.

With that I think I thank you, all of you, (who have) joined this MOOC and please do provide your feedback and I will try to improve it further. I will actually record all lectures afresh next time, whenever this MOOC is going to be offered, to make it more effective. We will have one more lecture where I will actually talk about the complete summary of whatever we have done in the whole MOOC.