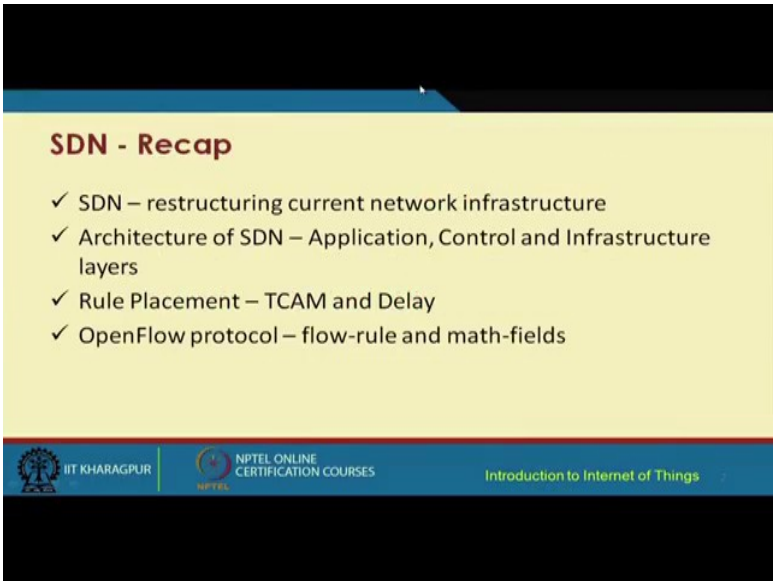
Introduction to Internet of Things
Prof. Sudip Misra
Department of Computer Science and Engineering
Indian Institute of Technology, Kharagpur

Lecture - 34
Software-Defined Networking - Part- II

Now, let us continue our discussions from what where we were in the first part of SDN. So, we are now in the second lecture second part of software defined networks. So, we have already talked about we have understood the rule placement problem and what are the different issues surrounding rule placement and TCAM; TCAM memory, it is limitations and how the rules can be created and how much time these rules are going to be maintained at the switches in the different-different flow tables of these switches.

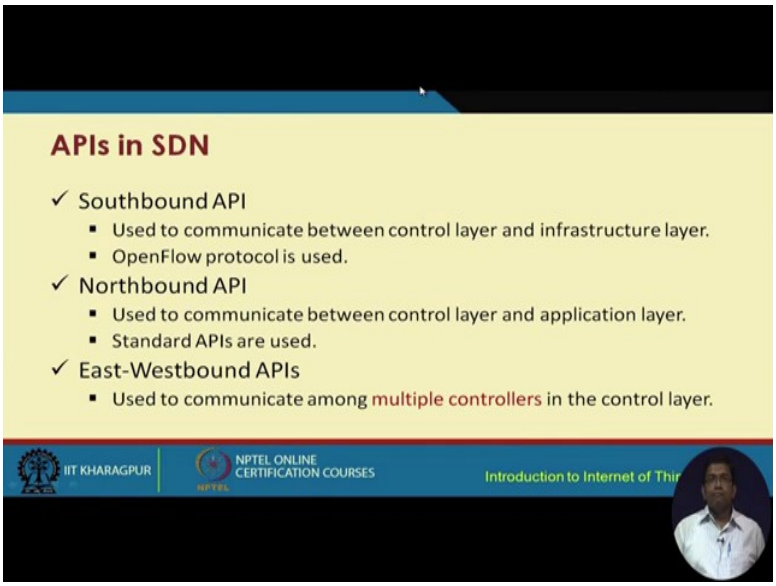
So, the other issue that is very important is the controller placement problem and this is what we are going to discuss next.

(Refer Slide Time: 01:01)



So, we have just as a recap we have already in the first part of SDN. We have understood the basic concept behind SDN the architecture of SDN the rule placement problem and the tradeoff between TCAM limitations limited memory TCAM memory and the delay. And we have also understood the open flow protocol and flow rule.

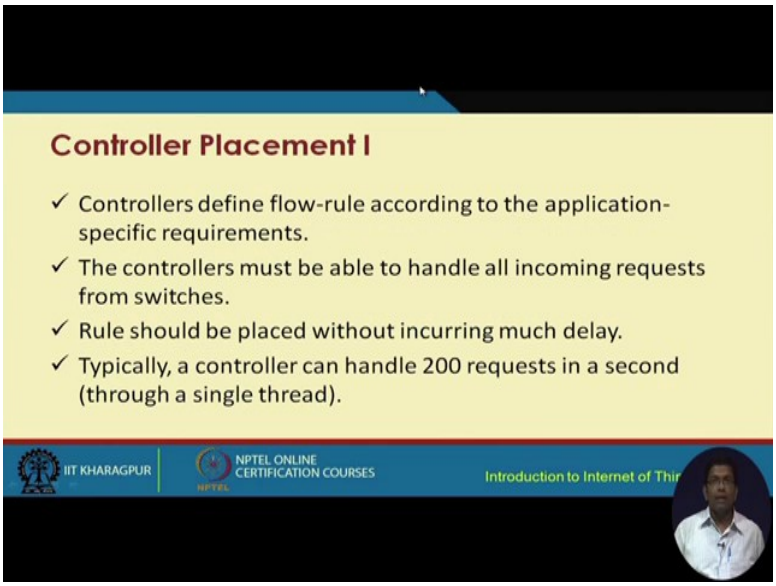
(Refer Slide Time: 01:32)



And the math rule math fields in the open flow protocol. So, in SDN between the data plane and between the data plane and the control plane we have an API and another API between the control plane and the application plane.

So, the first one; that means, between the control plane and the data plane and; that means, the infrastructure what we have is the south southbound API. And the open flow protocol is the protocol that is used for the southbound API communication for the northbound API this northbound API is used to communicate between the control layer and the application layer these 2 planes control plane and the application plane. And the all these existing standard API is that are used can still be used in the northbound API there is also the concept of another kind of API which is called the east westbound east westbound API and this concept this terminology comes into picture when we are talking about not a single controller, but multiple controllers in the control layer.

(Refer Slide Time: 02:41)

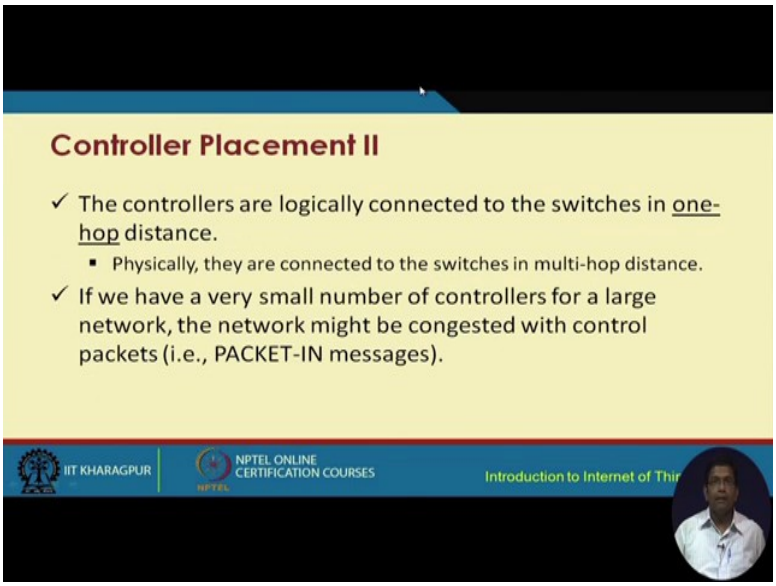


So, controllers basically define the rule the flow rule according to the application specific requirements. So, basically you know what is happening over here in SDN is that the control logic is basically you know taken care of by the control plane. So, you know control logic is separated out and as a result the controller knows what has to be done with a particular flow. And the controller basically controls the overall flow in the network and each of these switches what they are going to do that is done by the controller.

So, now, going back the controllers define the flow rule according to the application specific requirements the controller must be able to handle all these incoming requests from the switches and the rules should be placed without incurring much delay and that is important. Typically, in the rules the placement of the rules if it is not already existing in the flow table of a switch typically as we have seen before takes about 3 to 5 milliseconds, but you know this has to be minimized because otherwise what is going to happen is there are you know there will be a huge control overhead and that has to be that, because the controller is being separated out the control logic is separated out to the controller we do not want too much of a control delay control linked delay to happen there is a trade off definitely we cannot avoid this delay, but we have to minimize this particular delay.

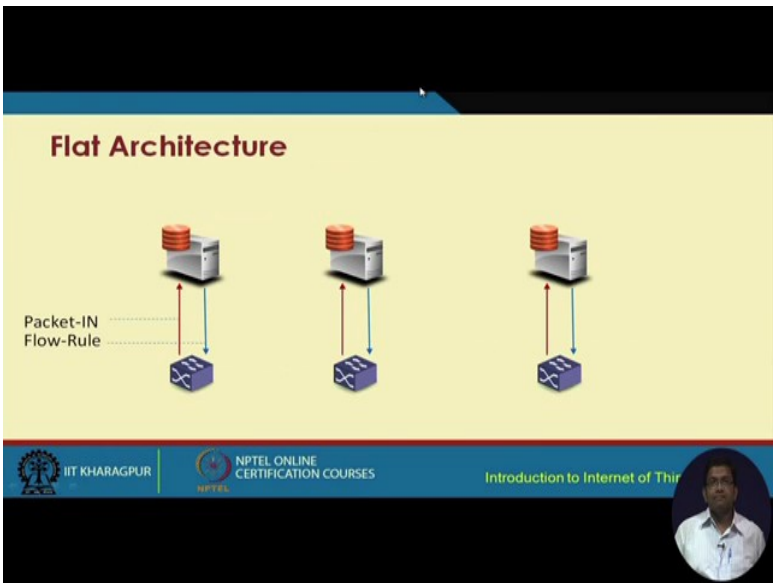
So, typically a controller can handle roughly about 2 hundred requests in a second and that is applicable for only the single threaded applications thing.

(Refer Slide Time: 04:24)



So, the controllers which are implementing similar thread, but currently multi threaded applications in controllers are also possible. The controllers are logically connected to the switches in a one-hop distance it is just a logical connection. So, the controller from the switch the switch thinks that the controller is away just a hop distance from it just one hop from it, but actually it is not so; actually you know physically when we are talking about the controller and the switch can be multiple hops away and typically they are multiple hops away.

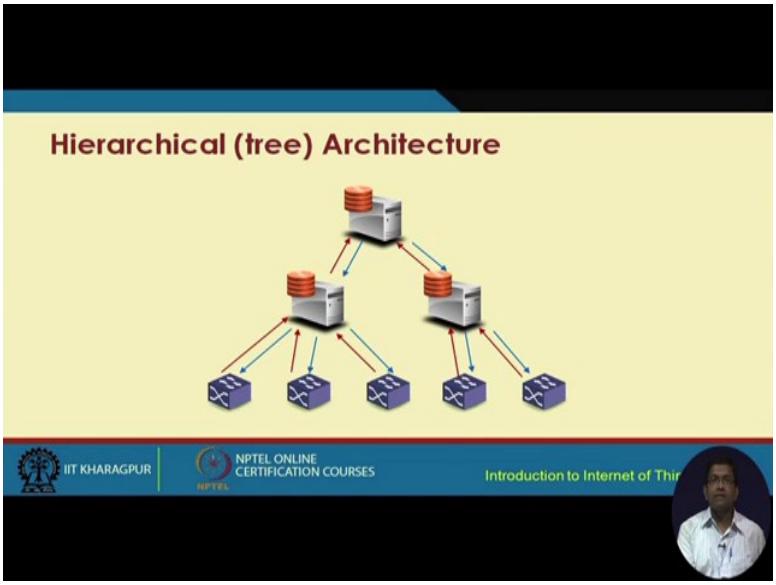
(Refer Slide Time: 05:03)



So, if we have a small number of controllers for a large network the network might be congested with control packets these packet in messages and so, if we look at this controller placement there are different architectures you know how and where we are going to place this controller. So, one architecture the basic architecture is called the flat architecture, and here basically the switch and the controller they are just logically one hop away the switch sends a packet in message to the controller if the switch already does not have this flow rule for the particular flow that it has received.

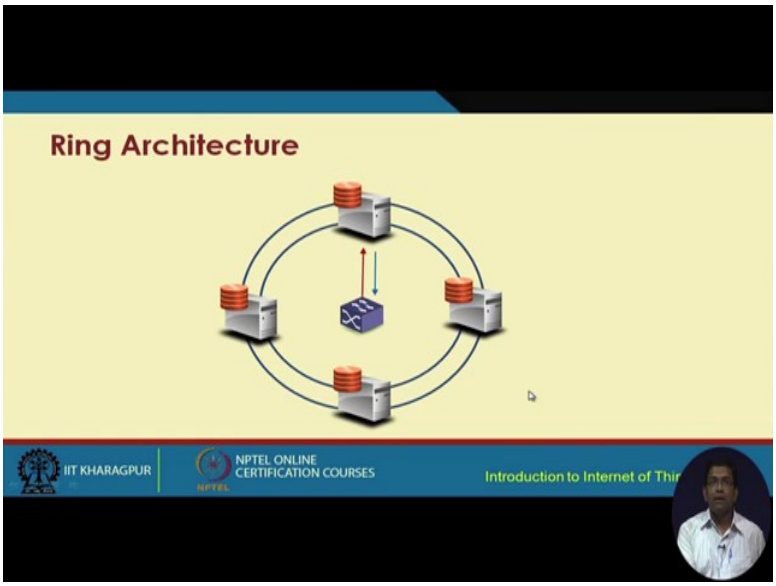
So, it will set a send a packet in message to the controller and the controller is going to send back the flow rule corresponding to that to that particular. That means, how it is at how the switch is going to treat it you know that particular instruction is going to be sent by the controller the controller knows it the controller knows how the different flows how the different packets are going to be handled this is the assumption in this particular technology SDN technology.

(Refer Slide Time: 06:09)



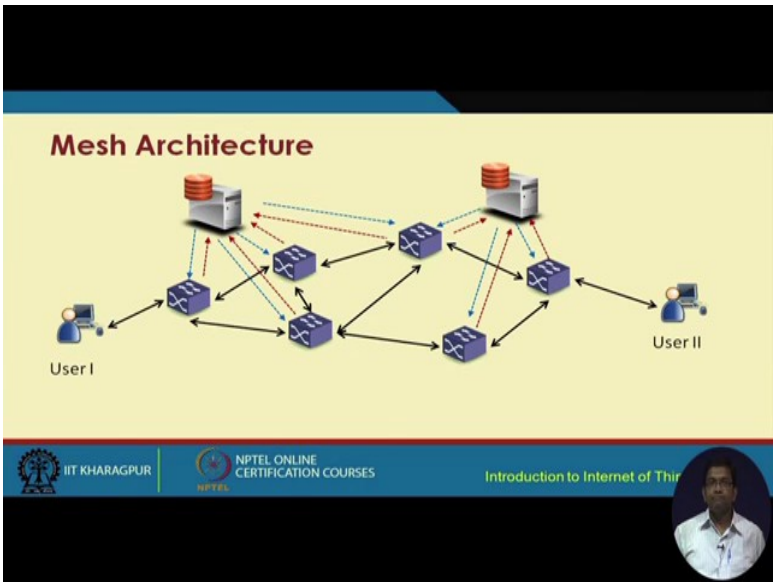
So, this is the flat architecture then we can have different other architectures this is the hierarchical or the tree architecture and these I do not need to elaborate further, but it is quite obvious we have these different switches and hierarchically they are placed within the controllers are placed and connected to these different switches in a tree like fashion.

(Refer Slide Time: 06:33)



And we have this packet in message and the corresponding flow rule coming back for each of these connectivity's then we have the ring architecture. In the ring architecture we have a similar kind of thing, but we have to keep in mind that in the ring architecture. So, basically these controllers are placed in a ring like fashion we have multiple controllers like this placed in ring like fashion, but a particular switch is connected to only one controller in this version.

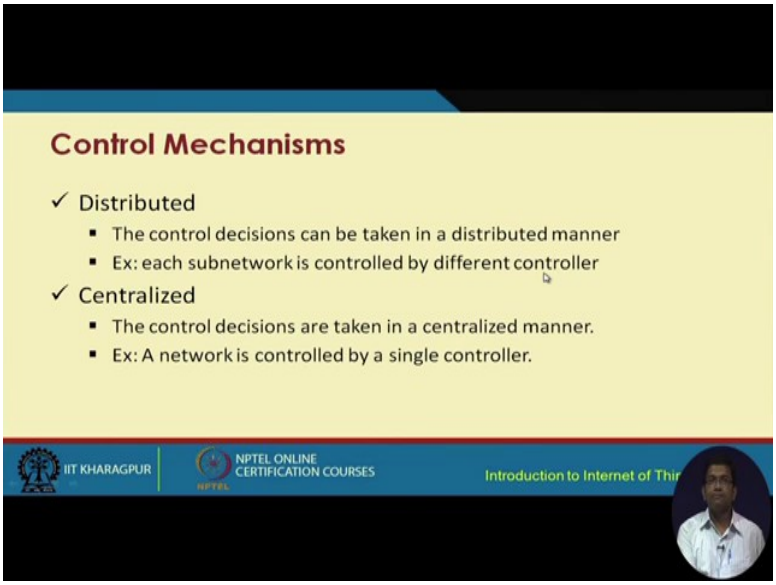
(Refer Slide Time: 07:18)



So, when the packet in request has to be sent this PACKET-IN request will be sent to a single controller only not that it can be sent to any of the other controllers in the ring it will be sent

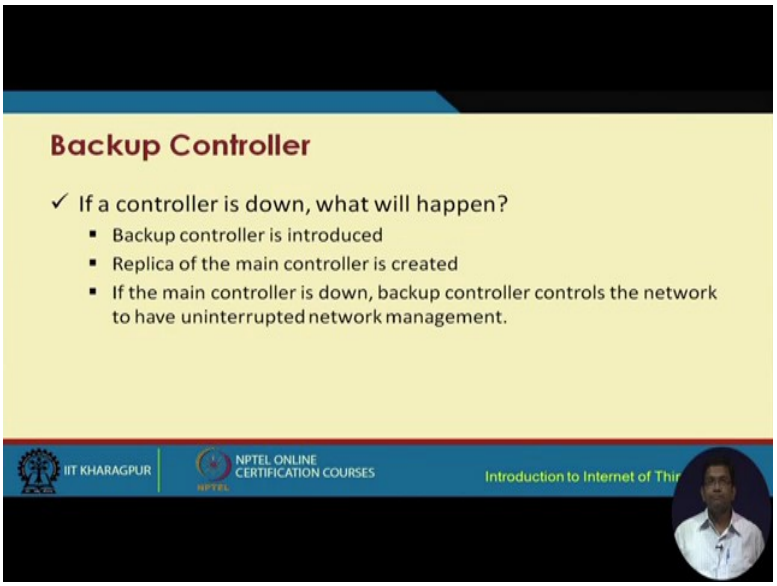
to a single controller and the flow rule is going to be sent to this particular switch that has requested the rule, and then we have the mesh architecture mesh as we know increases the reliability. And as you can see over here for instance we have 2 different switches who can be connected to a single controller. So, if this one goes down there is the other one which can take over and so on.

(Refer Slide Time: 07:41)



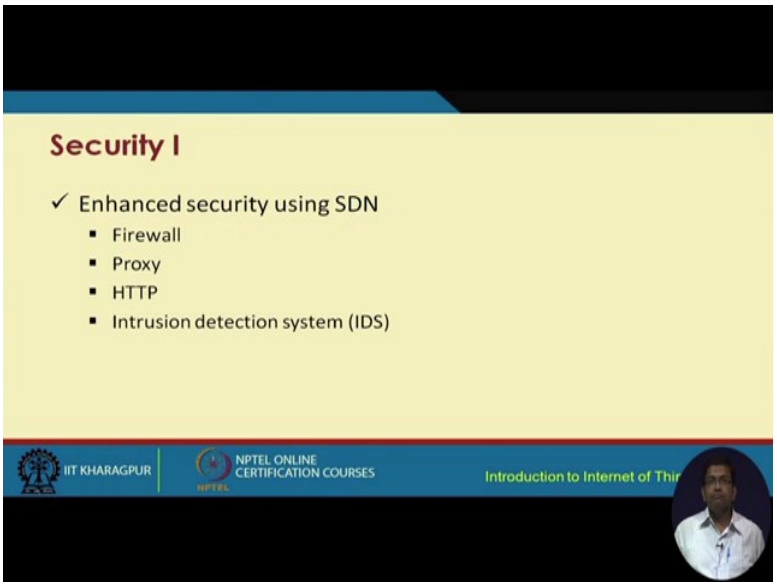
So, it basically increases improves the fault tolerance and improves the reliability of the network the mesh architecture. Now this control how this control is going to be done, how the control decisions are going to be made there are 2 different approaches one is the distributed the other one is centralized. In the distributed the control divisions can be taken in a distributed manner. For example, each sub network is controlled by a different controller. And in the centralized mechanism the control decisions are taken in a centralized manner. For example, a network is controlled by a single controller.

(Refer Slide Time: 08:21)



So, in such a case it is a centralized control and dividing into different sub networks are having a controller corresponding to it will be distributed solution there is also a concept of backup controller. So, if the primary controller is down then the backup controller takes over.

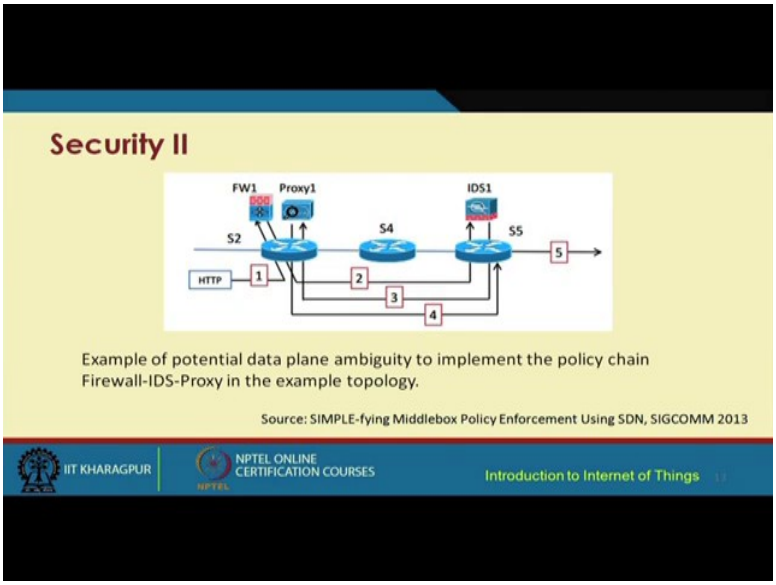
(Refer Slide Time: 08:42)



So, backup controller has a replica of the main controller and if the main controller is down the backup controller controls the network to have uninterrupted network management another very important thing is true in SDN one can have enhanced level of security in the

network and in this particular case we will be taking help of the firewall proxy http etcetera, etcetera and the IDS and these can have improved security with respect to this technology.

(Refer Slide Time: 09:05)

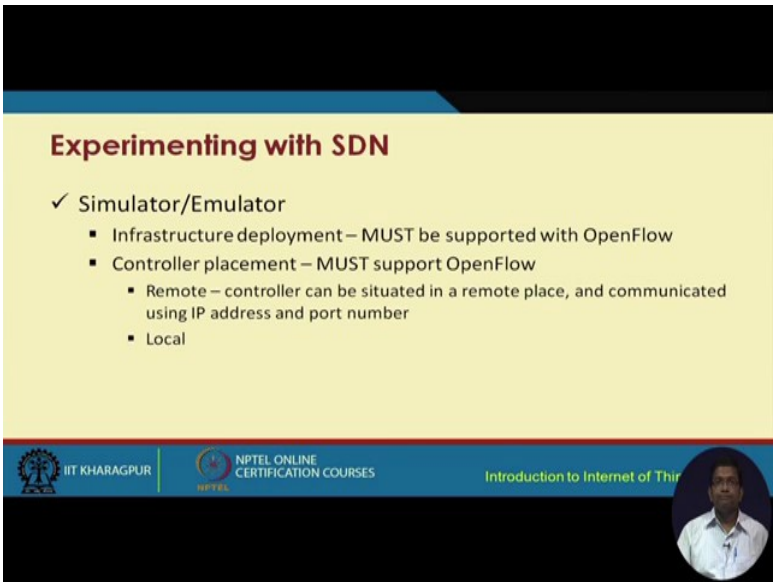


So, just as a very brief you know here we are not going to discuss about you know improving security with SDN and in much detail, but just as a brief you know this is the this is this is a this is a paper which was published in SIGCOMM in 2013 very recently; that means, which is talking about the simplifying protocol for policy enforcement.

So, what it does? So, you know let us look at this particular figure. So, it is an example of a potential data plane ambiguity to implement the policy chain this chain firewall IDS proxy in this particular topology and the sequence of flow of instructions is like this. So, it will this is from the http when a http request comes then it is sent from one switch to another switch. This particular switch then it goes to the IDS comes back goes to the proxy and the forwarding and the firewall.

And then finally, to this particular switch and then to the then finally, out of the network. So, this is how you know security is implemented and enhanced using SDN. So, we are not talking about as I mentioned already I just wanted to show you that security can indeed be improved with the help of SDN. And we do not want to discuss anything further on this particular issue.

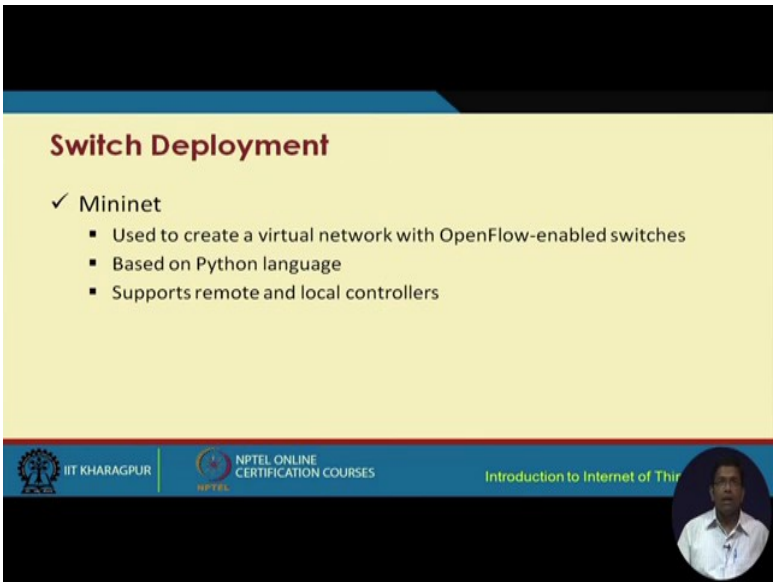
(Refer Slide Time: 10:39)



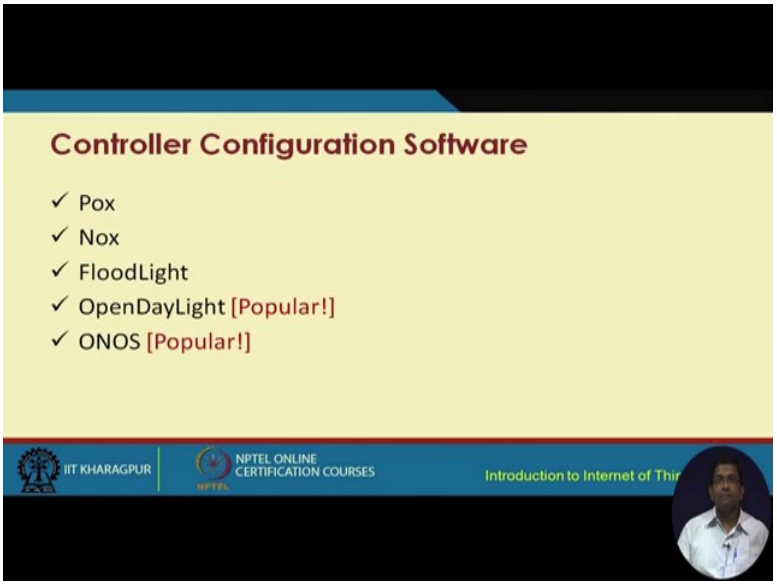
So, how do you. So, let us say that we have implemented SDN or we want to implement SDN and. So, so one of the things is that we have to experiment we have to experiment we have to evaluate the performance through experimentation. So, experimentation can be done with the help of simulators or they can be done with the help of emulators in emulators, basically what happens it is little different from the simulators. So, here the real emulators can be connected to the real network where real traffic can flow through this emulator and the data can be analyzed the performance of the network can be analyzed using these emulators. Simulators on the other hand basically simulate the entire thing packet flows, the network, the nodes, everything is simulated.

So, these simulators or the emulators take care of few different things one is the infrastructure deployment and this must be supported with open flow and the controller placement which must support open flow this controller can be a remote controller or a local controller and in the remote basically the controller can be situated in a remote place and communicated using IP address and port number a local controller basically does you know everything is local the controller itself is local and you know it takes care of it in a takes care of it locally for switch deployment the Mininet software is very Mininet simulator it is an emulator as well. So, it is very useful it is used to create a virtual network with open flow enable switches it runs on python and supports remote and local controllers both remote controller and local controllers are supported by Mininet.

(Refer Slide Time: 12:18)

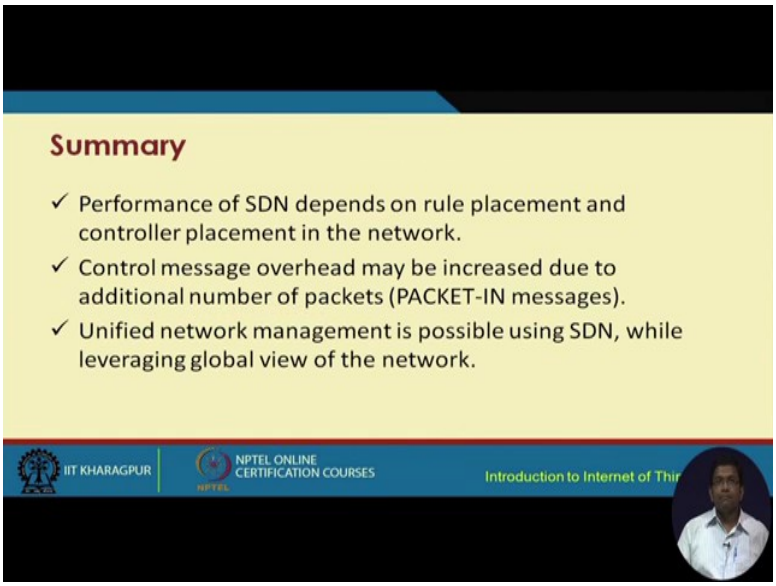


(Refer Slide Time: 12:45)



There are controller configuration software for example, Pox, Nox, Floodlight open daylight and ONOS particularly open daylight and ONOS are the most popular once that are used for controller configuration.

(Refer Slide Time: 13:02)



So, with this we come to almost the end of the lecture on SDN and we have understood the basics of SDN what is the motivation behind SDN why do we want to separate out the control logic from the data logic data plane which takes care of functions like forwarding etcetera to make the network efficient, but there is a trade off as well and we have seen that that more number of control packets are going to flow through the network.

But we want to minimize that control logic and this is where lot of research is being undertaken at present and we have also seen that the performance of SDN basically depends on 2 particular issues the rule placement and the controller placement in the network. Control message overhead as I was mentioning before will be increased due to additional number of packets the packet in messages from the switches to the controller going through. And what is required is to have an unified network management to be made should be made possible using SDN while leveraging the global view of the network. So, we need to have an unified unified network management architectural framework using SDN with this we come to an end of the SDN lecture.

And so, next what we are going to talk about in another the next lecture is how SDN can be used in IoT to make IoT efficient internet of things efficient. So, we will have to wait for the next lecture in order to understand this.

Thank you.