Foundations of Cyber Physical Systems

Prof. Soumyajit Dey

Department of Computer Science and Engineering

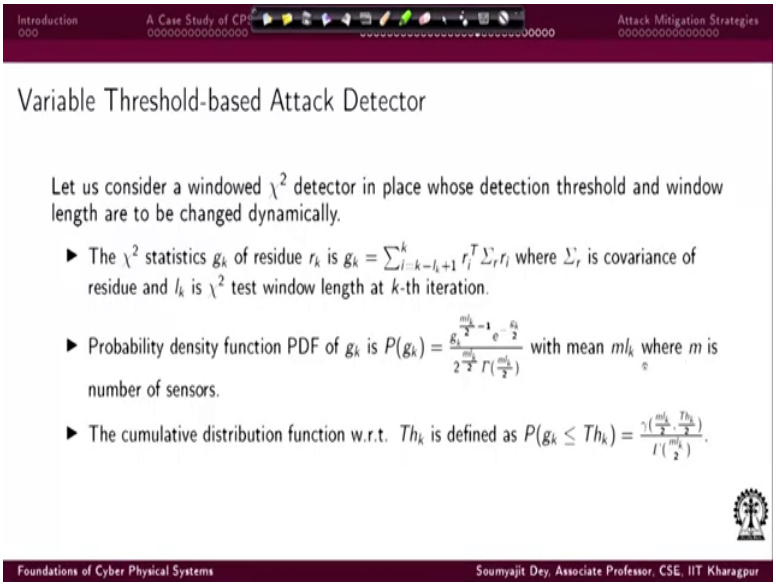
Indian Institute of Technology – Kharagpur

Lecture – 57

Attack Detection and Mitigation in CPS (Continued)

Hello and welcome back to this lecture series on Foundations of Cyber Physical Systems. So, we have been talking about these ideas on variable threshold-based detectors. Right So, we said that well it should be possible to change the detectors parameters based on the space in the trajectory where we are working ah so that the attack detection accuracy increases. Right.

(Refer Slide Time: 00:51)



So, ah if we consider this windowed chi square detector. The parameters of this detector was this threshold that we have been setting. And also ah the scale length window ah inside which this chi square statistics was being computed. If you remember this was my chi square statistics, where sigma r is the covariance of the residue and l_k is the chi square window, the test window over which I am summing up, this different residue values.

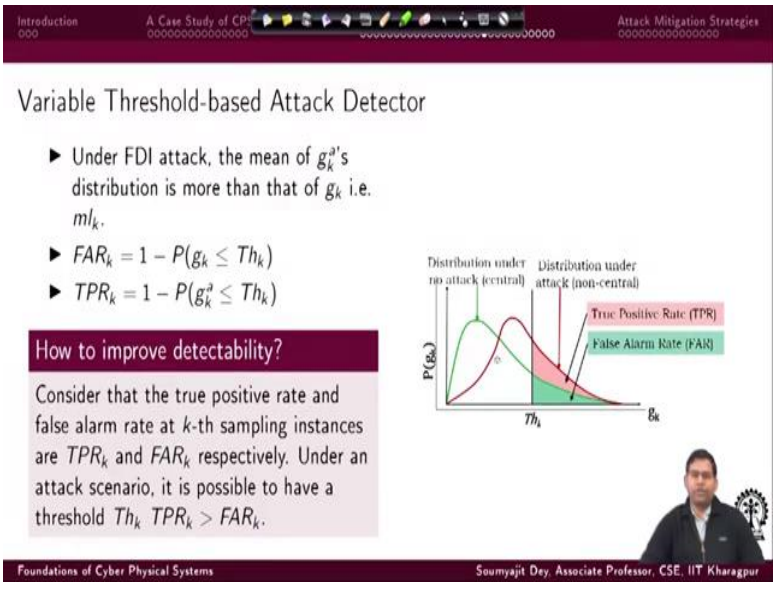
Now, ah from the mathematical machinery of chi square, ah we can create this probability density function ah where, ah we can say that well it is distributed with respect to this Ah probabilities I mean the distribution follows this expression and the mean is m times l k where

m is the number of sensors we are considering and l_k is in the kth instant. What is the ah window size of the detector we are considering?

And from this PDF we can actually integrate it and figure out, ah integrate up to the threshold value ah for this kth iteration whatever threshold we are setting for the control systems, kth iteration. And we can actually figure out what is the CDF the cumulative distribution function. That means it is the probability that this chi square stat is less than the threshold and we can I mean from from basic I mean ah statistics we can see that well this would be this kind of a nice well formed function where where these are the gamma functions. Ok Now, ah the when we will cover these things, a bit more detailed in our assignments and tutorials, where we will actually give examples of well how these things really work. ah For here for now, let us lets just consider it in a formula ah in in The form in this form, as is given here.

So, these are the gamma functions and you can see the parameters are the threshold, the number of sensors m and the and the ah window length on which I am computing, the chi square, stat here. Ok.

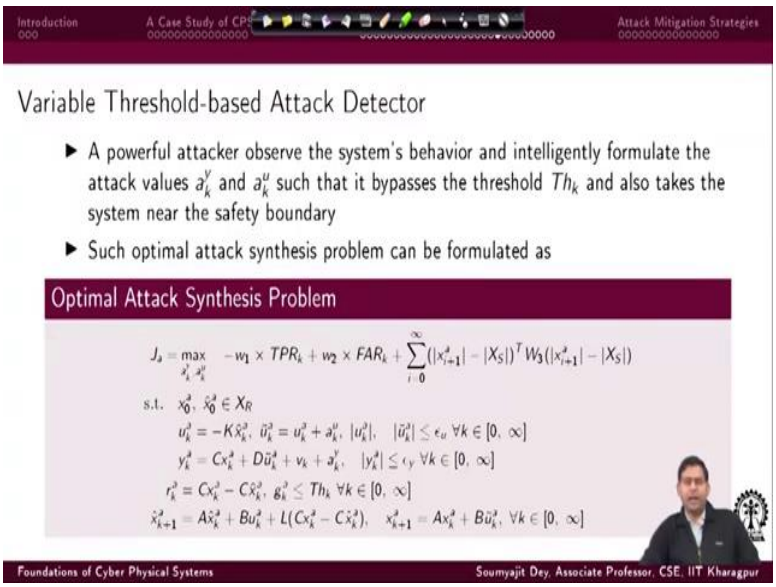
(Refer Slide Time: 02:46)



Now, this thing we have already discussed that well ah since I know this probability so, suppose there is no attack happening. So, 1 minus this that is the probability that g_k is greater than threshold, is basically the probability of false alarm. And when, ah there is an attack really happening ah then 1 minus of this is the probability that there is a true positivity. right Now, this distribution, ah this distribution actually shifts it is ah and becomes a non-central distribution when there is no attack.

This can be proved we are actually omitting the proof here. All I am trying to say is let us just correlate these expressions here ah with this This picture here. So, essentially when I say that what is the probability that g k is greater than threshold is nothing but the integration of this curve from threshold point up to infinity. right And if the distribution is the green one when there is no attack then you have this green part and when there is an attack then this distribution is for the true positive rate. And since the distribution is shifting here right ah to a non-central one it is going to be it is this part. right Now, the question is ah well what should be my target when I am trying to decide, I mean create a detector. I should I should try to create a detector who is chooses it is parameters of threshold its parameter of window length in such a way that this thing happens. That means the true positive rates value ah is higher than the false alarm rates value.

(Refer Slide Time: 04:24)



So that based on that actually we created this optimal threshold synthesis problem. And we say that well what should be my parameters of this detector so that this objective function gets satisfied. And accordingly, you can also create well this is my detector design problem and if somebody is trying to design the attacker. Then what should be the goal of the attacker?

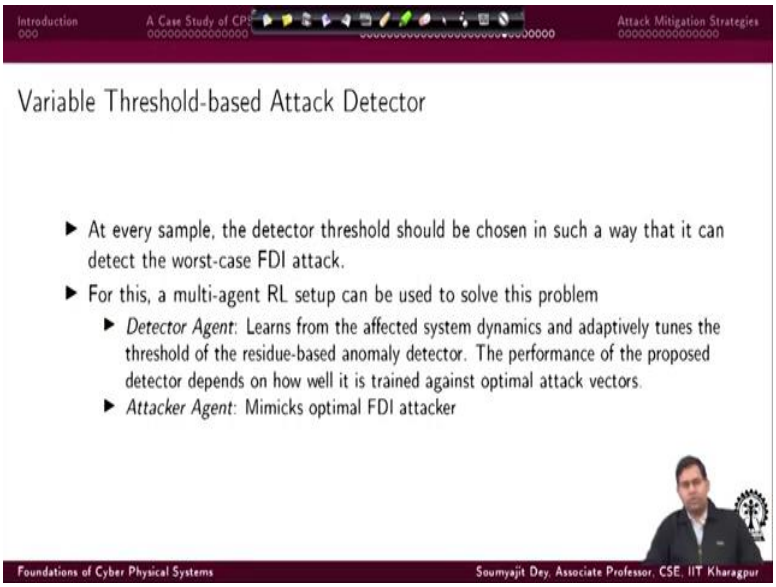
The attacker's goal should be that well I want a lower true positive rate, a higher false alarm rate and a higher estimation value. I mean I have higher value of ah I mean estimation error of the state. Right So, suppose you are trying to create an attacker model and you are trying to say that what is the attack? What is the amount of attack? ah That this attacker is going to inject. So, it can be found by maximizing this cost function.

Where what is this cost function doing? It has made suitable choice of W_1, W_2 and W_3 these weights. right So, these are scalar weights and this is the matrix best weight. OK And what it is doing is well it is saying that we will choose this in a way that so that this W_1s, this W's are positive and positive definite here. So that the attacker will try to give weightage to decreasing TPR increasing false alarm Ah and increasing the estimation error here. Right

So that is how the objective function is designed. And of course, all these two choice of the attack values have to be such that you satisfy these constraints. right Because these are the standard control system constants you have for the dynamics of the system. So, while satisfying all these constraints, if you can choose suitable values so that this cost function is maximized in every ah kth instance then you can have a suitable value of the attack vectors which a try which have a higher probability of bypassing the detector. Right So, ah actually that is how it works because ah um based on this, so that means, ah you have to solve this complex constraint satisfaction problem in every iteration of the control system. That means in every iteration you have to set up this objective function.

And solve with respect to with respect to all these constraints and accordingly, you get the values of attack values on the measurement and the control input. Ah Now, the thing is this is the complex situation. Right I mean solving these has to be done in real time and accordingly, the attacks has to be injected with almost no delay. That is virtually impossible.

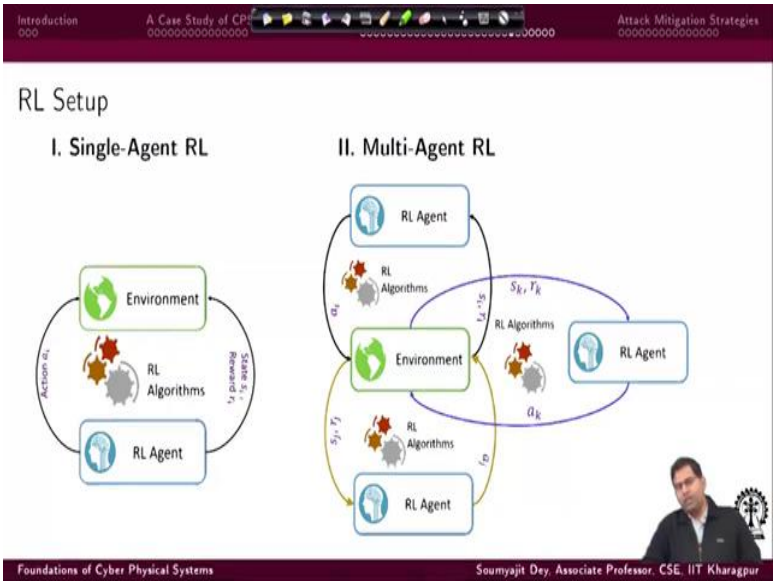
(Refer Slide Time: 06:57)



So, ah that is why we I mean we said that well let us have an AI agent which learns how to attack and which also learns how to detect. So that means, if you if you are trying to create an AI agent which is trying to learn a detector, you should make the detector learn over and attacker agent's attacks also. right That means you should create a nice attacker agents and you should see that the attack agents goal is to ah disturb the function of the system.

And the detector should be able to learn what should be the threshold based on which it can detect the attacker. This attacker agents ah attacks that they are causing. And in case this detector agent, is able to detect the attacks. Then ah it can be deployable and in a deployed scenario it can actually subvert Ah other attacks also. Right

(Refer Slide Time: 07:50)

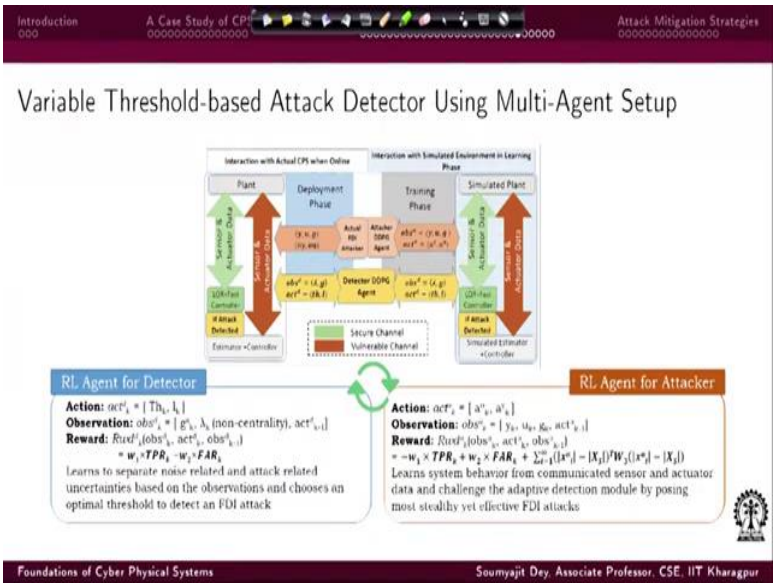


So, we actually formulated it as a multi-agent reinforced learning system. So, as we know that a reinforcement learning system ah is basically a very popular AI technique where you are trying to learn ah some way to give actuations to the environment. But the environmental model is quite unknown to you. So, is like thinking that well I am trying to design a controller for a plant for which I do not really have a model.

That is why RL based techniques have found in recent times, lot of applications in complex control design problems where you do not really have a nice model of the plant. right So, ah just in a nutshell, what RL agent is supposed to do is? It will it is trying to test an action and accordingly, it will receive a state update that will the environment changes or state from this to this and you get this kind of reward. Right

And if there are multiple agents, each of them have their own actions and based on that they can observe their rewards. And accordingly, the agents learn that well what should be it is optimal action so that the rewards are kind of maximized. So, in this case, if I try to map it. For an attacker is trying to generate actions like a_y and a_u and what it is trying to observe as reward is well what is the maximum amount of error I am able to induce into the estimation and whether I am able to create a stealthy attacks in in the process.

(Refer Slide Time: 09:19)



So, based on that you can ah just create this kind of agents. Ah One agent for the attacker and one agent from the detector, as you can see that this attacker, what it is going it is trying several is going to try some attacker actions initially. ah And then it you will it will observe well what is the measurement? What is the control input? What is the chi square statistics, etcetera, etcetera. And accordingly, it will evaluate this cost function that we design.

At this cost function is rewarding the positivity negative. I mean decreasing of true positivity and increasing of false alarm rate and increasing values of the estimation error because this is from the attacker side view. If you see the detectors cost function, what are the detectors choices? The detector is trying to set suitable parameters for the Ah detector. That means the agent for the detector is trying to set suitable parameters.

That means it is trying to choose a Th_k and a l_k so that ah after it observes that one closed loop control iteration, it is trying to figure out whether whether what happened to the system? What is my calculator TPR and what is my calculated FAR? Based on the previous formulas we

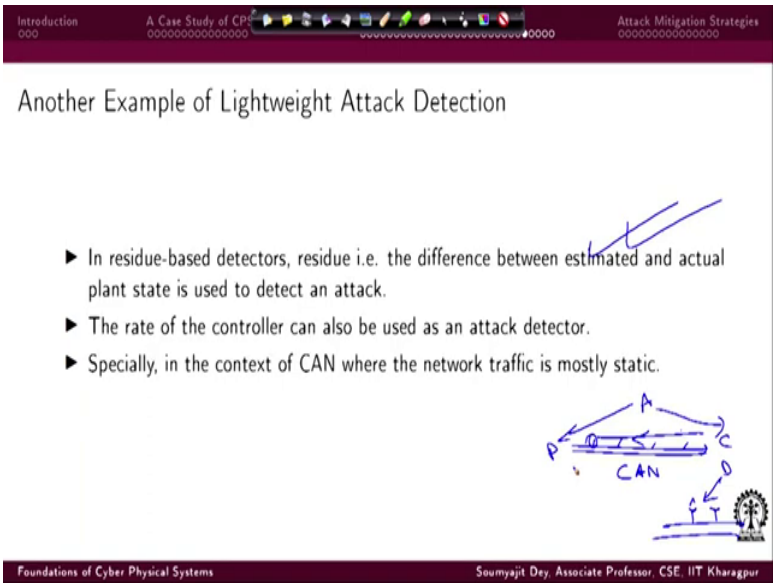
discussed and and if there is a gain in TPR and a decrement in FAR, it is going to reward it as per this formula. Right So, this is how, ah the idea will work here.

It will try to maximize ah it will try to reward the detector ah RL agent if it is able to increase, TPR and decrease FAR. And the agent for the attacker it will try to it will give awards to the attacker if it is ah doing the opposite and also in the process it is going to increase the estimation error because that is the objective of that attacker. right Now, if this game is played between the agents in a simulated environment.

We actually have examples of how to create some simulated examples in open AI gym and similar simulators. But that is an advanced stuff it will be only ah I mean we are not going into the details here. We are just trying to motivate you to think that this is indeed possible. This kind of CPS security problems can be mapped to AI problems and if you are interested, you can actually model it in AI simulation environment. Ah

So that that is how a control problem can be mapped to an AI simulation environment and you can learn control actions. You can learn ah protection methods that well for the system for which I do not have an attack model. If I use this kind of an attack agent model ah where I know the reward function and stuff, what will be the lethal that the attacker agent can then learn. And based on those later attacks, how they can be prevented that the detected agent can learn. So that that is the point we are trying to make here ah through this example.

(Refer Slide Time: 12:11)



The next thing we talked about was this CAN based attack detection techniques. right ah If you remember, we talked about what I mean what are the ways in which CAN buses can be actually attacked? Right CAN based systems where you have multiple ECUs, they can be attacked and well ah 1 ECU can be thrown the victim is you can be thrown to a bus off mode. right Now, the thing is well we if we have residue base detectors now.

Let us think that you have a control system implemented over a CAN. That means the packets, the measurements and the control actions are moving over the CAN bus as a packet. right So, ah what I mean in in our previous example of residue base detectors what we did? We were computing, this estimates and the estimate error and we are trying to detect whether an attack has happened or not.

Now, there is something different we can also do based on the architecture of the system. So, your architecture of the system is where you have a CAN here, you have a plant here you have a controller. In our previous system that is this we have the controller and also a detector and whenever your measurement came here ah you fired up the detector earlier. And you calculated an expected measurement value. Right

And you saw that well, what is the measurement you are getting and you do a difference here and see that will ah if that estimation, error is higher from the threshold based on chi square statistics or non-base detector, whatever you are using. And you say that well there has been attack or not. But you can do something different. In this model you are assuming that the attacker, the attacker, knows what is the plant?

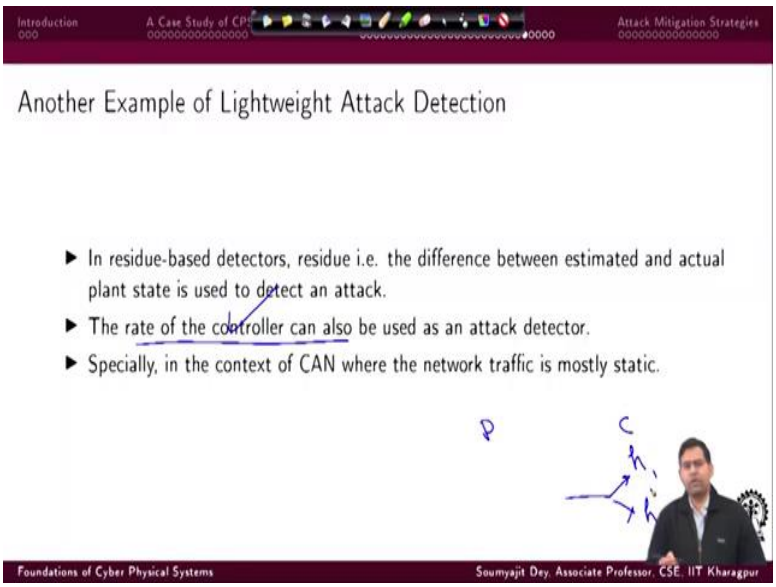
What is the controller? What are their periodicities? And exactly when these values that means the plant measurements are moving from the plant to the controller. And exactly what is the period and offset with which Ah these values from the control controller to the, to the plant, those control inputs are moving. So that is a that can be a very ambitious attack model, ah which may not be true.

Alternately, what can happen is something like this that well um suppose I decide that well I will change the rate of the controller. And this is something unknown to the attacker or the plant and the controller has a mutually agreed upon schedule. And it in that schedule it is

dropping the measurement communication or the control input communication in such in certain, certain cycles.

If that is the case then if an attack is happening at those points, they will be detected. Because we get a value at a point when I am I am not I am I am checking and I am figuring out that there is a CAN message with an ID. ah Because of the attack but I this this message was not supposed to be there because the controller knows. This is not a time when that value is supposed to come. So, if a controller and a plant decides on such aperiodic schedules then ah there would be some ways in which the attacks can be detected here.

(Refer Slide Time: 15:25)

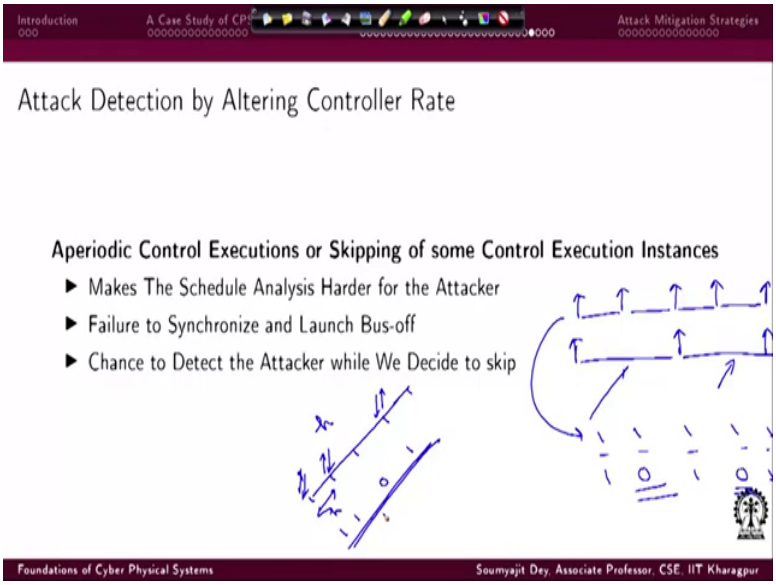


So, let us see such an example. So, this is what we call aperiodic control execution or skipping of some control execution rates. So, whatever we decide here discuss here mind this that they can also be achieved by having a controller which has a modified rate that an adaptive rate. That means you have the plant, you have the controller, let us say the controller is sampling at h_1 now.

And then suddenly, the plant the plant is being sampled at h_1 and then at h_2 and accordingly, with that in synchronicity the control updates rate is also changing. Now, if the switch of rates is not known to the attacker or that occur is unable to guess that. right So then that can also be a way to ah detect attacks. Because suppose I suddenly change the rate from h_1 to h_2. I know that then at a modified offset and timing, the CAN packets are supposed to come.

But if the attacker does not know that then he will be sending the attack packets exactly at this period and they will be detected. Because those are so, basically there is a timing-based detection. You know that at these points no CAN packet is supposed to come but if I get a packet that means there is an adversary existing in the network.

(Refer Slide Time: 16:39)



So, let us talk in general about aperiodic execution or skipping of some control execution, ah which I said earlier that at certain points of time, if you understand that well this change of rate can also be modelled as aperiodic execution. ah So, what I mean is let us say I am working with a sampling rate of ah h and suddenly I increase that rate to 2h then that but if the attacker does not know this then he is attacking here and here and he is getting caught.

This situation of modifying sampling rates in general can also be talked about as skipping of control execution. So, let us say I do not change the rate. But rather this is like a sequence of 1s Ah the positions where I am sending the controls. For later, I started introducing some skips. That means I started saying that well in this next control cycle I do not send the control packet or I do not use the measurement. Similarly, here stuff like that. right So, if you do such kind of skipping then what you have is an aperiodic control execution.

That means let us say I decide that well I am operating at this period h. This is all h but like I said here, I take the measurement and the and the update, here I take the measurement and the update here I do not do anything here again, I take the measurement and the update.

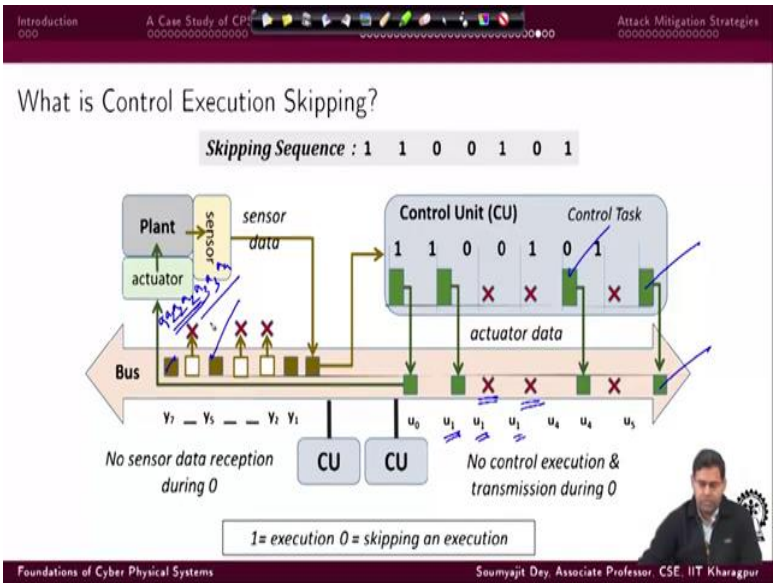
So, it is like a binary pattern of 1 1 0 1 and I am saying that where I will repeat the pattern. That means I am saying that well inside the pattern I am I am not doing I am I am skipping the control execution and I am skipping the measurement gathering process in the every third instance of a 4 length ah 4 length pattern. So, this is what is our idea of skipping control executions that we decide that there are certain points when I will periodically change ah I will just drop the control action. Ok

And I will just also drop the measurement ah measurement because if I do not know compute control execution, I do not need the measurement. But the thing is the trick is that the attacker should not know this. ok So then for the attacker, analyzing the schedule and identifying when to attack becomes difficult.

So, because the attacker does not really because you see in in case of bus off the attacker, has to analyze the schedule and figure out when the controller or the plant is sending a measurement or actuation packet. right Now, if occasionally, I have this kind of drafts ah this kind of drops then for the attacker figuring out Ah those attack points is immensely difficult. right Ah So then in that case there will be there will be failure to synchronize and launch bus off. And also, even if they are able to do that ah the attacker may resort to sending packets at positions.

Where the controller and the plant knows that there should not be any packets because we have decided to skip those positions.

(Refer Slide Time: 19:55)



So, let us take this example. Suppose we have this kind of a skipping sequence. That means we have decided that well we will ah we will execute control here, here but not at these points and then again at this points. ok So that means I am taking some measurements on the bus. OK um So, the way we are showing this thing is, we are kind of ah we have this packet here there is a measurement packet.

And accordingly, ah control action is computed that is kind of denoted by this 1. And accordingly, ah control input value is again sent as a CAN payload over the bus. So, this is the control unit. This is your plant actuator sensor system. So, the sensor senses this value, so, let us say this is y_1. And accordingly, Ah the controller sends it this a_1 because the y_1 has been used. This packet has been consumed there has been computation and an update here.

And then there has been y_2 and accordingly, this has also been computed right and this has also been sent so, you have a_2. right But then you decide not to send because your schedule says that well let us not do any communication in the next two cycles. So, there is no communication for y_3 and y_4. OK And accordingly, on the controller side, there is no reception and control input computation. right So, these things ah are not computed. Ok

So, accordingly, I have I have a_2 holding up here right the actuation or let us call it ah I mean, if I just write it here in terms of the us this is my y_0. ok So, for a minute let us just use my notation that I have already here. right So, I have a y_0 and based on that you have u_0 and then you have computed u_1. But here you do not compute, u_1 or I mean u_2 or u_3. right So that means what you are doing is well you are just holding on with this u 1 here and here. Right

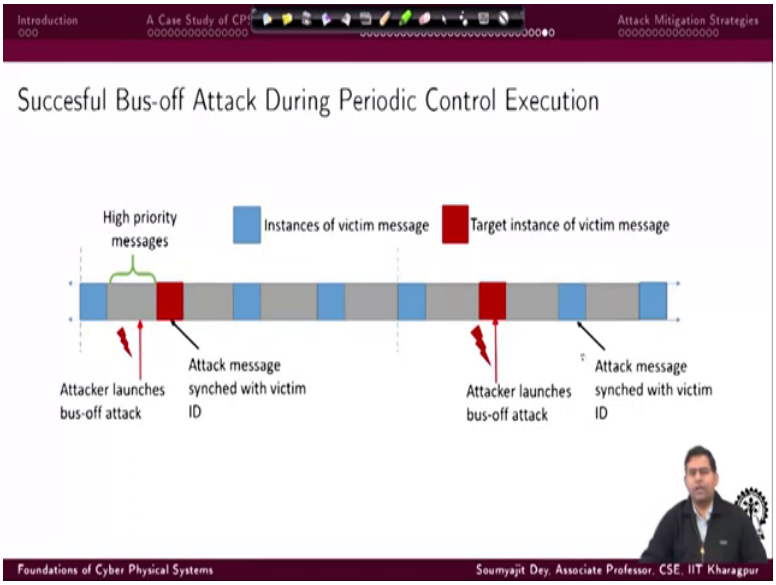
So, if I see here, I had my actuation a_1 then a_2. In the next two cycles I do not update the actuation it remains a_2 and a_2 only. ok So that is the thing I do not update the control. Now, again, I have a 1 here, so that means I will take some value here. Accordingly, I will do some control input calculation here. Right So that will update this. right A new controller has been calculated and the actuation is updated.

Again, maybe there is a skip, so, this a_3 will continue and then again there is actual computation with this measurement there is a computation. And there is an actual control action moving so, you have a_5. ok So, as you can see, ah what is happening is? ah You have skipped certain

communications on the bus, both from the plant to the controller and the controller to the plant. Ok

And we are representing this phenomenon is a shortened enough by using a binary stream. So, what this means is? There is no sensor data reception and there is no control execution and transmission during this thing. Ok

(Refer Slide Time: 23:36)



So, if we do this, ah what is the advantage? That is the thing. right The advantage, as we have been saying that there are two fold advantages. One is the attacker when he is doing a scheduled analysis of whatever CAN traffic you can see on the bus. And he is trying to guess that well where to attack. ah It will be difficult for him, because he will be having difficulty in figuring out exactly at what point the attack injection must be happening.

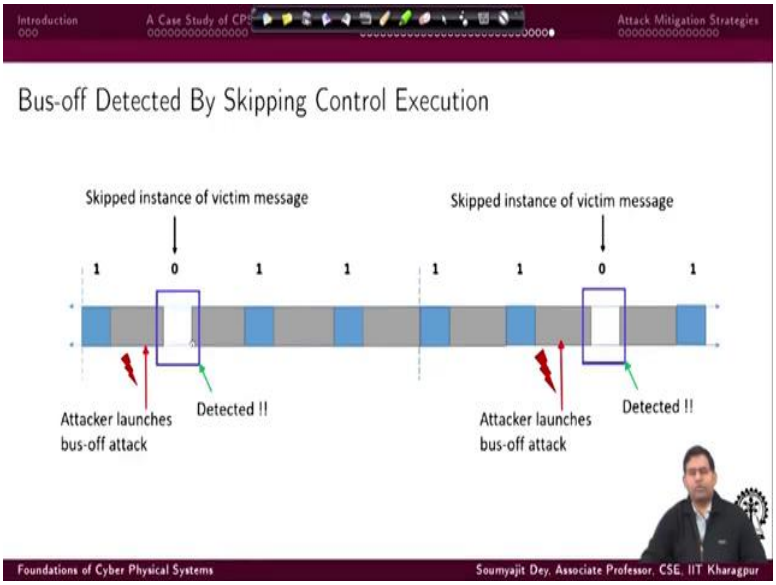
Now, they also are smart. So, what an attacker can do is attacker will launch an attack when a high priority message is on the move because that cannot be prevented. And then immediately after the high priority messages that attacker message is queued. And then it will be transmitted in sync in the next period, with the victim's message. right So that is how the bus off attack works that you target a high priority message.

The high priority message will not let you go but you will be queued. And that means now you have a synchronization point because you know that the moment this high priority has gone. ah the The victim's message which has same priority as mine whenever that will get access to

the bus, I will also get access to the bus. So, let us say here ah you have the attack message which is synced with the victim.

And these are the instances of the victim message and this and you are again targeting ah this victim message here and you are launching the bus off attack. So, what will happen is eventually the attack message will get synced with the victim message.

(Refer Slide Time: 25:07)



Now, ah when you have this skip instances right. So, let us say before Ah this skip instance the attacker has queued a message with respect to high priority message which was here. So, due to this high priority message the attackers message could not be sent but it has got queued. right So, the moment this message finishes the attacker's message now goes but on the controller it will receive this message. right because

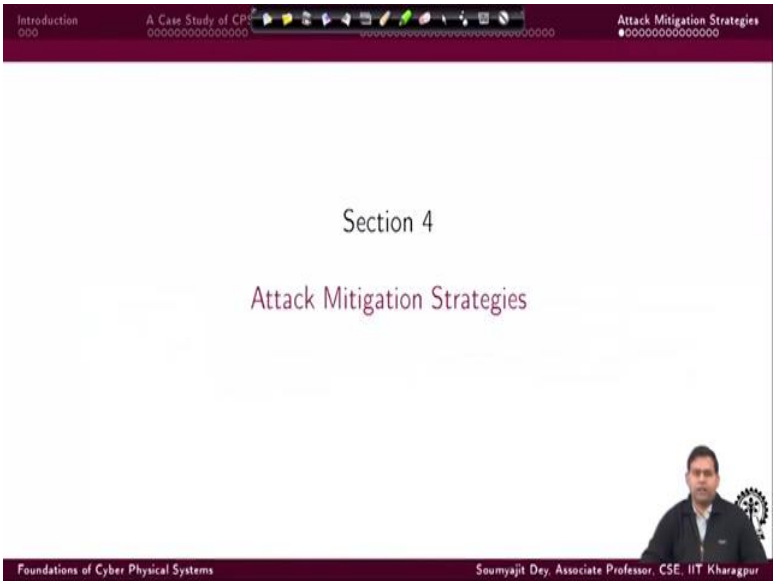
But it knows that at this point I am not supposed to send any message. I am not supposed to receive any message. Right So, it will detect that there is an attack happening. So, ah that is an import, that is a nice technique through which attacks can be detected in a system by ah actually deciding that well I am not going to send control updates in a regular way. I can either make the controller aperiodic that means the controller switches it is period.

And according the control gain value after sometimes which have been pre-decided and the attacker does not know that. So, when the attacker will continue in the old rate old speed of control ah eventually it will it will be it will be detected or we do not change the controllers

period. But what we do is at certain periods or certain cycles ah we are not sending any message at all that is pre-decided.

But now when the attacker is going to send a message and that is the only message that will get queued because the victim is not sending anything. So, it will be detected. So that is that is one way to look at it.

(Refer Slide Time: 26:39)



Now, with this maybe we will end this class. And we will resume again with our next topic, ah in week 12 which is attack mitigation strategies. Thank you for your attention.