00:00:04 good evening everyone and welcome to the next lecture uh we were discussing about virtualization we continuing from the last Tuesday's class uh this Saturday class and the coming up Saturday class are special classes optional for regular but there we are discussing more about handson practices uh so if you remember we left that uh there are two types of virtualization OS level virtualization and Hardware level virtualization and the question I gave is to find out the reason which one is more performance efficient and why why

00:00:46 was more important I kind of answered you also that hardware virtualization is more friendly for performance purpose whereas OS level will have more performance overhead so let's continue our discussion so why do you think hardware virtualization is more efficient it has been seven days you are living with this question I hope yes anyone would like to share so what's the question so the question is why hardware virtualization is better than OS level virtualization in terms of performance performance because in OS

00:01:40 level virtualization we the uh the kernel is shared okay but in the hardware level the kernel is not shared so you are saying here kernel is shared in the OS level virtualization correct you can see The annotation right online people yes sir it is yes sir uh it is shared why it is affecting performance uh because the same kernel is doing all the like see uh in the OS level the same kernel is doing all the uh calculations or whatever the interpretation is going on but when we do the hardware level virtualization the

00:02:26 different different kernels are being hosted on the same machine machine and they are doing the processes it is becoming bottleneck ah yes we can say that all guest operating systems are using the same kernel the kernel become bottleneck yes now uh look at this uh argument like uh say in your laptop you have is one good option sor because of hardware virtualization power consumption is why power consumption is less because single Hardware is hosting multiple OS so only one Hardware is used for this let come to that okay first let's

00:03:21 discuss about whether Kel is really a bottleneck or not say your system has 8GB RAM and it has say uh i7 processor and it has 256 GB hardk and some Network B we are saying if kernel is given this complete Hardware it won't be able to handle this Hardware won't be able to operate this Hardware simultaneously because if you want to use this complete Hardware through different guest processor then your kernel is becoming bottleneck all the requests are going through same kernel okay uh in general this water neck is not

00:04:29 there for uh in general this bottl neck is not there from software side because here we are talking about a single physical machine and if software is becoming bottleneck to you especially the operating system is becoming bottleneck to that Hardware means that operating system is not suitable for that Hardware specification right for example uh say you have 8GB RAM and you have Windows XP I hope all of you have heard this term Windows XP it used to be an operating system the famous one and it can only

00:05:23 support okay uh I'll come to that what it can support but with Windows XP with 8 GB Ram you cannot open a file which is of 6 GB you cannot open with Windows XP any reason why XP does not support 8GB Ram uh no it supports 8GB Ram because most most part of R is occupied to open load this file so it's need some another spaces for execution also read and write and the look and the front front all the supports so uh physical classroom answer is coming that uh Windows XP used to bring the complete file into RAM and

00:06:27 then open it but nowadays partial file is also okay to open by Advanced operating systems but you have 8GB Ram the file size only 6gb less than 1 GB Windows XP Windows XP requires somewh around 900 MB RAM active Ram to run how many of you are studying or studied computer architecture you should be able to answer so why Windows XP is unable to open a 6gb size file on say if 8 GB you are feeling it's less you say it's 16 GB even on 16 GB Ram physical system Windows XP won't be able to open a file

00:07:25 which is 6 GB of size so your movies you won't be able to run on Windows XP the movies which you are uh if you are downloading and watching then it generally goes more than 4 GB 5 GB right you tell me uh is it because 32bit uh architecture or file system what 32bit architecture is stopping you so the ram limitation happens over there there some answer coming from physical classro see uh both the arguments are coming that the ram may not be accessible above 4GB or something that's not the reason reason is

00:08:40 nearby 32 okay what what is limitation but uh uh so Windows XP support NTFS file system also Windows XP extension supports NTFS but originally it was only f 32 and uh here the reason mentioned correctly by physical classroom person that pet 32 can only support a file up to 4 GB that is true so 6gb file you cannot open because our file system is not support now if you have say 16GB RAM and Windows XP with pad 32 kind of file system you can say this uh physic you know this Hardware specification is not

00:09:34 compatible with the software you are running you can run more advanced software but the reason you gave about the kernel being becoming Bal n is not the reason most of advanced operating systems are able to uh handle the hardware specification on which they are running so what could be the reason uh for which OS level virtualization is getting slower as compared to Hardware level virtualization sorry uh context switching between processes uh gets complicated Contex switching becomes complic because the resources are not

00:10:32 distributed so what do you think how much time in contact switching is happening and General in a CPU cycle yeah I mean someone has blocked uh the resource for sometime and because it's not uh predefined resource blocking also happens here yeah but it's predefined and distributed yeah here also it's predefined and distributed the size of your VM if you are allocating 2 GB or 6gb or 4 GB that is redefined reallocated in both the cases okay and regarding that contact switching maybe out of 10 or 20 CPU

00:11:21 Cycles one cycle is being spend on contact switching so the performance overhead through contact switching is might be uh 1 by 20 time not more than that right so that's uh okay but if we discuss about OS and Hardware level virtualization OS level virtualization is visibly slower as compared to Hardware level virtualization when I say visibly means you can feel that now your operating system is slowing down if you use o level virtualization some answer is coming from physical classroom uh I'll come to you Rees so uh

00:12:19 his answer is in Hardware level virtualization Hardware is optimized for virtualization task in no level virtualization first operating system is generic operating system but at the end your Operating System is using the hardware where operating system should know how to schedule a job on CPU how to block registers how to create a program control block PCB and how to navigate through Ram that's a basic requirement for any operating system to execute any program right same happen depends on hypervisor

00:13:01 host OS also yes rites you are saying something Sir with the hint that you have given it is visible that uh there is a requirement of or there is a operating system the host operating system operating to handle the request between the guest OS and the hardware in uh normal scenario and uh OS uh there is no direct interaction with Hardware directly if you are talking about my hint uh I gave hint in the last class more layers means more performance overhead now you should justify me why there are more layers in OS level

00:13:57 virtualization I have deliberately shown this diagram in this way sir because uh where you have written host OS the host OS uh on top of it will be the guest OS that are supposed to be there they will only interact with the host OS and that will handle the commands or whatever is being given to the hardware whereas in the hardware virtualization the guest OS uh will be allowed to uh directly interact with the hardware the latency is supposed to reduce interacting with Hardware that's what you are saying

00:14:39 right yes sir okay uh yes an India sir number of layers are more in case of hardware virtualization then hardware virtualization should be slower right yeah it means Technically when ever we have more layered slower than Hardware layer that's what our point of discussion is so even if I put guest operating system also on top of horse to which I have mentioned here in both cases so where hardware virtualization is getting faster as compared to S level virtualization yes when you talk about Deadlocks uh

00:15:47 he's saying that deadlock probability of Deadlock occurrence is higher in OS level virtualization so when we talk about Deadlocks Deadlocks happens on a resources now both Hardware are of same specification and on both of on top of both we are running two VMS so essentially two VMS are sharing that Hardware specification in both the cases so dead log occurrence chances are also similar it's not same as of now it's similar what you can see so one basic question what is like what is difference between host o and guest

00:16:32 o host o is running on the hardware the be metal and Gest TOS is the operating system which you install inside virtual machine this means like in hardware virtualization we it's a 2os is running in both the cases two os are running one is host one is guest sir but uh in the left case that is OS level virtualization the host OS is a standard OS which is not made to or built for handling these requests whereas to the right the hypervisor we are saying it is hostos but ultimately it is a software or tool built just for this purpose so

00:17:19 it is this layer is bound to be negligible what is operating system what you something that go ahead that converts our human level things to human level and uh commands to uh Machine level commands that is compiler and assembler it's a platform between hardware and software yeah please be more elaborate like it will give the resources available for the software to execute any process if you want to execute it will give you all the CPU cycle hardware and registers uh Supply to you and the memory Supply so that you

00:18:08 can execute your process operating system does the resource management resources sources based on your requirement based on the processes requirement okay anything else uh can we call it as an interface uh where the different devices connected to the computer can be managed or interacted using a software the main purpose of operating system to provide the resources to process so that it can be execute okay so uh I'll come to the traditional definition where we say operating system connects various

00:18:51 external devices connects them and made it available for users but then with this definition any other applications also becomes operating system for example your security camera app it connects your mobile phone and it connects your maybe some security camera and it works well and similarly your other applications like uh uh using external camera with your mobile phone that application also connects external devices is and ensure that your you can work through your application so that was traditional definition we used

00:19:36 to study in primary school but when you do operating system course in your btech you understand that it's all about Process Management file management memory management secondary storage management uh concurrence management right it's all about resource management and then we study about Deadlocks so if there's a race around condition many processes are asking for same resources can we allocate to all of them or we should allocate one by one if we allocate some resources to some processes and other resources to other

00:20:13 process and these some and other processes keep waiting for each other and then deadlock happens right so if it's all about resource management and that resource management is also being done by host operating system here and host operating system hypervisor here so ultimately your resource management task is being done by hor to as always in both the cases I'm coming back to the question now why there are more layers in OS level virtualization so virtual box is uh here on see virtual box and then on below

00:21:16 that you have horse device right I'll draw it again maybe more clearly you have host to layer then you have say virtual box layer and then you have Gest OS and at the bottom you have for to clear now I think we have more layers in operating system level all the layers are revealed now now still explain me how does it making it uh slower because virtualization box layer will come into effect at the time of machine configuration when you are creating a virtual machine at that time you decide using

00:22:13 virtual box that okay let's allocate 4 GB Ram two processes uh processors and uh maybe 20gb hard disk to this VM this Gest operating system VM so virtual box will play active role at the time of configuration or starting the machine once your virtual machine is running then it's it bypasses this virtual works and directly connects to host operating system for most of tasks for scheduling processes running any application or software directly connects to host operating system with its upper limit as

00:23:02 4GB Ram or whatever hard disk we have allocated essentially your virtual machine in both the cases becomes a process for host operating system how many you have used Java I don't know if you have run very large program in Java and it has exceeded 5 12 MD Ram generally program holds because your jvm to use maximum 5 12 MB RAM but you can go to your jbm configuration and change it and then you can run large programs what is jvm Java virtual machine it's a virtual machine and we are talking about virtual

00:23:55 machines only there's no difference between that J VM and the VM which we are talking about both are virtual machine both are given some resources so when my VM or my guest process is given 4GB upper Ram cab it can use up to 4GB Ram so it will directly go to host operating system as a process where it can use up to 4GB RAM and it will use it that run time where the additional year is coming all of you are very near to answer uh because in OS level we are using some name spaces like Docker and all those stuff they are adding the

00:24:54 during the they're doing the isolation and adding the layer so uh doer we are not discussing at this point of time doer is a different technology it's a container based it turns only on host operating system it can it doesn't have any version like hardware virtualization we have OS level containerization only like doers we don't have Hardware level containerization as so let's come back to virtualization or hyper visors so why OS level virtualization is slow now answer should come from your operating system course how many of you

00:25:47 have done operating system course people are now not raising hands I know half of you have done operating system course please answer why OS level virtualization is slower answer is in what is operating system okay anything related to paging no don't say critical section next and so so okay what constitutes an operating system in general layers are increasing that's why system operating system provides a kind of abstraction or layer of abstraction so in the first case if you see so O Level virtualization the the

00:26:42 layers of abstractions are more compared to the hardware virtualizer so that's why it will be slower explain this abstraction layer a bit more what operating system abstracts out suppose at the core we have Hardware so above the hardware generally we studied in the operating system so there is a kind of layer for managing the resources operating system knows what kind of Hardware are there within the system so that kind of layer is getting increased in the first case so that may be the reason of lower performance you are near to

00:27:21 answer but not at done sir means like uh in Virtual ition we have like fixed location of the memory CPU or anything whereas in local o we can increase we have we can use the cloud you can assume that efficiency it's fixed for both yes sir in W level virtualization means all the processes are sending it requested to uh host was so host was actually handle all the request so but in case of hardware virtualization all the guest ways are directly uh interacting with Manus Hardware so uh when you say uh all the

00:28:15 request what that request does with operating system say you want to open a browser how browser opens up in through operating system how does operating system helps so there will be more system calls or functions calls so that's why from one layer to the another can you justify your argument why there will be more system calls because going from one layer to the another suppose from the guest process we go to the host operating system then the VM then the host host operating system of the hardware so uh

00:28:56 one of the con constituent for operating system is the libraries which operating system comes with like Dynamic link library for Windows or uh processor block proc files Etc files in Linux you can see them clearly so when you run anything at application layer it generates some system calls at operating system level now those operating system calls needs to be translated which is compar uh to the level of host operating system uh like I don't know how many of you have tried creating virtual machine

00:29:47 you can create a virtual machine you can have a say Windows laptop and you can create a Linux VM right in Virtual box if you have a Windows laptop you can create a Linux VM now system calls of Linux are different from Windows operating system right so your application is running in Linux VM your applic a will translate your system calls into Linux system calls and then this virtualization layer will come which will translate this system codes to Windows system calls and then it will run on Hardware so

00:30:59 uh you have Linux VM system calls of Linux are getting converted to equivalent system calls of Windows and then those system calls are running on Hardware whereas in case of this this host operating system which is also known as hypervisor it's a bare minimum operating system which you require to just to boot up the machine and allocate the resources in a virtualized form your device libraries or dynamic link libraries are not part of this hypervisor so here you have only dll here in the guest operating

00:31:55 system whereas in this kind of architecture you have dll here with guest operating system and you have dll here in host operating system so uh when you translate now system calls it goes here then this now hypervisor layer will not interfere here it's kind of bypassed and directly run on Hardware now although hardware virtualization is giving us benefit in terms of performance because there is less layers and the second is your host operating system has occupied uh few resources your host operating system is

00:32:52 running also running so it requires some memory it requires some hard disk it requires some Network with so it has occupied some resources so now it leads to another scenario another uh discussion Point say you have 8 GB [Music] Ram how many VMS of 4GB Ram you can create how many VMS create a 4 GB Ram with 8GB Ram physical system scenario one for OS level and scenario two for Hardware level one so logically we can create means more than one but we cannot run simultaneously even logically also we

00:33:47 cannot create more than one in case of os level because logically or even without logic also they operating system is occupying say 1 to 2 gb Ram what is remaining Available to You 6 GB Ram right so you can if you create a VM of 4GB Ram the remaining another VM can maximum have 2 G approximately we can create a two machines VMS of 2GB and 4GB right but so when we create a VM so we just create a simple configurations right means uh so creating a VM uh just hold on uh I think you are here for fairwell that's in

00:34:52 308 so today there's a fairwell of register uh and that's happening in our class which is in 308 so we are moved to 110 and some people came here to attend fairwell so I send them back okay okay yeah we are coming back to class yes you were saying something and yes sir so when we create a VM so at that time means we are just creating a some configurations right means we are not actually allocating anything uh to the physical when we run that particular VM then those according to that particular specification things

00:35:32 I agree with your statement so what is's saying that theoretically we can create infinite number of VMS with 8GB Ram uh we can create uh any number of VMS up to 6 GB or 7 GB Ram allocation we can get 10 VMS but we cannot run them Sim simultaneously at run time your memory is allocated to VM all of you are getting the point so I'll change my question uh this how many VMS you can run simultaneously of 4GB on a 8GB physical machine answer is only 1 VM another VM will not have full 4GB available because

00:36:26 your host operating system has occupied some Ram at run time what about second case with hypervisor uh Hardware level virtualization how many VMS of 4GB Ram we can run with 8GB physical machine still one because still maybe even 500 or 200 mbam is required by hypervisor but we are saying hypervisor will occupy less memory space because it's not a full-fledged operating system in fact vij here has also mentioned in the comments saying that general purpose OS has support for all sorts of device drivers whereas hypervisor may not have

00:37:26 support for a special device drivers for example in early days when Windows XP was there and if you plug a new pen drive in Windows XP you need to reboot Windows XP in order to use that new pen drive or new company new vendor pen drive it might sound strange for this generation you can plug in any pen drive from any company your system does not require restart but Windows XP used to require restart even if I bring a new mouse and plug into my CPU Windows XP used to require restart if I attach a new dash cam

00:38:17 webcam to my uh Windows XP I require restart what is the reason yeah because you have to load the uh device driver for that specific uh yes so this D concept Dynamic link Library Concept in Windows came much later after Windows XP uh I think Windows V came up with this or what came after Windows XP Windows Windows V only Windows Vista then Windows 7 came yes the Windows Vista came with this concept of D and Linux were already using dlls they were not calling it dll but this Dynamic link libraries are the libraries which can be

00:39:15 dynamically linked to your execution at runtime so your new pen drive or your new mouse or new keyboard or new webcam require new device driver and Windows XP was not supporting this Dynamic linking of these libraries so you need to restart your system then whatever device drivers you have installed it will be linked and then you can use new pen driver or new webcam but in case of Windows VRA and 7 at the run time they can dynamically link the library so what happens a new device plugs in or motherboard sends request

00:39:59 that okay for this new device I do not have uh compatible device driver I should look into secondary storage it looks into secondary storage it found new device driver installed it picks from there and uh bring into RAM and then use it but Windows XP was not such Dynamic case but sir D were present in the Windows XP and previous version also even in Windows 98 uh those dll only for the uh libraries which are coming by default with new device drivers you need to restart but a DL concept was there then

00:40:46 why the restart was required because they were not using for new user installed drivers they were only using it for operating system drivers in that sense your operating system was not adaptable for new device drivers but once you sto it and once you restart then it will always come then you don't require dll was by that time it becomes your part of your configur D okay so there are different types of uh loading uh compile time loading just I'll write here one is compile time loading then run time

00:41:49 loading so in compile time loading means what you can change those variables in your program only at the compile time or before compiling and if you want to modify those variables you need to compile it again right at in runtime loading you can change the value of variable at runtime and it will be adopted dynamically so when I say you need to restart your Windows XP with new device drivers what time loading Windows XP is using might we compile time were you compiling Windows XP we were not compiling Windows XP we

00:42:39 were restarting so something is there in between compiling and runtime ver boot time loading yeah bootstrapping boot time loading means what [Music] you uh assign a value to any variable at the time of booting after that it will be fixed so your page size in your operating system is boot time variable or compile time variable compile time time variable what is general page size in your operating [Music] system what is the page size of your operating system one homework for all of you do it surely this time format your

00:43:54 friends laptop because they don't know about p size of your operating system their operating system what is the P size in your laptop in your operating system 4K almost typically it is 4K 4K b or 8 KB one of these two so it's a compile time variable right so when you bring ISO file which is compiled code or page size is already fixed there and if you want to change your pce size in operating system we need to do the changes in the code of operating system compile it again create an ISO file bootable file

00:44:48 and then boot it those who are feeling excited about changing page size they can try installing Arch Linux Arch Linux so source code is available you can change the source code you can write your own device drivers also in large Linux and then compile it and then install it the best idea is to install it it at the hardware level the second best is you try installing it in your virtual machine so now P size is compile time variable your device drivers which are being linked in Windows expera was boot time

00:45:43 variables but with newer Windows they become runtime variables this something else between boot time variable and runtime variable is another time of configuration I answered it it's configuration time loading which is clearly visible in your mobile applications when you install any new app you configure it at the beginning with your permissions whether location services are allowed or not microphone is allowed or not so it's a configuration time loading runtime loading is you don't need to go back to configuration again

00:46:35 and again I'm writing here so configuration time loading which is here between Boot and runtime so runtime parameters you can change at any point of time during your uses configuration time parameters you can only change at the time of configuration of course you can always reconfigure your applications no one is stopping you but that's a configuration level changes then boot time parameters can be fixed at the time of booting and compile time parameters you need to change them at compile time so

00:47:21 uh your answer about Windows XP is now clear for the pre installed preconfigured uh libraries it was able to bring them in memory at the run time but if a new library comes in new device driver comes in that need to be changed at the boot time now there's a concept of registry addit registry concept is there in Windows I don't know how many of you have used registry edit option anyone here so yes sir yeah yes sir I think what is the command R edit redit this is the command right yeah yeah R edit go into your command

00:48:17 terminal on Windows and type RG edit you'll find all your systems installation configurations are there in large number of parameters maybe few thousands hundreds parameters and your viruses your operating system viruses does some changes with RG added and may enable or disable some applications some permissions and all so you can also actually go into registry addit and if you have knowledge you can delete viruses from there you don't need antivirus some viruses are unable to detected by antivirus but with your

00:49:04 knowledge you can go to registry AIT and identify them and delete them don't go to your registry AIT today evening and delete anything else your operating system will become faulty might become faulty you will not realize and it will you know the fault might be visible during your course project demonstration so there's a relationship between the bug happens here and the error happens somewhere else right bug is not always the error error is something else bug is something else so uh registry I did in Windows XP

00:49:49 was boot time facility with Windows 7 and all it becomes runtime facility you can uh change value in registry edit so whenever you install any new device driver it changes in registry edit at the runtime and your operating system always uh look for registry edits the latest values okay uh so the first uh drawback you mentioned for this s level virtualization that will come into place only when you're having like to separate o right so this because of the system calls you will have different you will

00:50:31 have dll coming at two places what if the architecture is optimized and it knows that it will only run say only windows or only Linux OS in that context would it still be a disadvantage uh in that context for OS level yes because still your uh requests are converted from uh your application to system call in guest and those calls are getting converted into system calls of host operating system the conversion might be faster but layer is there it's not directly running on Hardware I'll I'll explain with

00:51:20 example U so there's a process management and there we have job scheduling qes right now we know in guest operating system there might be three processes running so those three processes might be in the que so after P1 P2 term will come after P2 P3 terms will come now here this job scheder is of guest OS for horse TOS this gas TOS is only one single process and these P1 P2 P3 are not visible to host operating system now host operating system might be running some other processes like H1 H2 H3 one of them will be of this gastro

00:52:26 say G1 another processes might be G2 now host operating system will schedule all these processes here in same CU and whenever g1's turn come on your CPU then G1 say get 10 millisecs time assuming now G1 in this 10 millisecond will schedule this P1 P2 P3 based on its own scheduling algorithm it can be round robin it can be priority it can be multiq whatever similarly for G2 whatever milliseconds G2 will get it will run its uh other guest processes right so this decision of scheduling your process

00:53:28 into a Q and keeping track whenever Q's turns comes then our program control block will be uh contact switching will happen your program control block will be saved for the existing process and will be new PCB will be brought for new process so that concept switching is happening at the OS level and this is also happening here at the between P1 and P2 contact switching so whenever ter of P1 goes and P2 comes contact switching happens in guest OS but when turns from G1 to H2 comes contact switching is happening at the

00:54:12 host o so your job scheduling part your contact switching part your PCB management part is happening twice for your final application which is you are running here right so that will happen in parallel in the guest OS in the hardware level so there is minimum dependency on the host that's what it does what happens in because it's a minimum hypervisor only which is only doing this kind of resource allocation level task your job scheduling can be directly run here with this 4GB allocation and its job scheduling

00:55:00 will run here with its own 4GB or 3gb allocation whatever BM is the size yes thanks so what what is happening here is suppose in your classroom which I uh uh say if some of you have some special requirement that top two two possibilities either I ask you to First Contact T and if they are unable to resolve you contact me or another possibility is you always contact me which is happening right now so I request the first one so what is happening is when there there are tear in between between your

00:55:49 request and my response they are creating some delays which you might feel right which is okay but it's not uh but if everything comes directly to me there might be cases of bottleneck but delay won't be there delay won't be there in case of virtual machines because we are only running those VMS which are fitting well within our Hardware specification because their Ram allocation especially the ram allocation happens in the beginning itself so your host operating system hypervisor will not allow you to run

00:56:37 more VMS with overc committed Ram one number of VMS with our committed so if you run uh 8GB machine you run 2 GB 3 VMS after that if you want to try running 4th PM your host operating system will say it cannot run because memory is already full or by chance if it allows you then you cannot use any VM or your host operating system because no process will find CPU or uh CPU to execute anything because Ram is to full more than 8GB you will not have contact switching algorithm running also in that case so your operating system or your

00:57:38 hypervisor will prevent from over commitment at run time but if you uh recall from uh Saturday's lecture where we talked about that uh your hard can be dynamically allocated for virtual machine your hard disk space can be dynamically allocated so there you can actually overcome it and then there will be uh race around condition uh I'll explain with that one scenario say you have 10 GB hard disk in total assume you have 10 GB hard disk and your host operating system is occupying say 2 GB hard disk space after installation

00:58:37 and all then remaining free space it 8GB I create four VMS and give them hard disk of 2 GB each now there are two options one is we allocate dynamically this hard space or statistically now if we allocate this hardest space statistically after creating this four G uh four VMS of 22 G if I come back to my host operating system check how much hard disk is free it will show me zero byte because 2 GB is occupied by itself the H operating system and all these 22 GBS freezed and allocated for VMS and actually even if you using say

00:59:58 uh tiny Os or Puppy Linux or mint OS which may not require 2 GB of hard disk they may occupy say 1 GB of hard disk for installation purpose so actually they might be occupying only 1 GB but because of this static allocation from virtual box or VMware or any hypervisor ler for host operating system it show that 2 GB is blocked and if we allocate dynamically whatever guest operating system or virtual machine is occupying the actual space say 1 GB in our scenario then at Host operating system layer it will show that 4GB is fre space

01:00:57 now you run this vm1 you install any more application here the size of 1 GB May extend to say 1.5 GB now the installation size of BM1 has become 1.5 GB now how much free space is available at Host operating system here 3.5 right I hope all of you are following me Al it's not a complex MTH yes any question any doubt here so if you choose an option to dynamically allocate hard disk it will only allocate the actual space required to for all these installation files for VN but if you use a static allocation whatever the maximum

01:01:56 allocation we have done it will create a dummy folder of 2GB size inside that folder it will be empty but the folder size to host operating system will be mentioned at 2GB I'm using folder term essentially your VM is also a folder essentially your device drivers are also a folder or file so everything in your operating system is is a folder or file nothing else whether it's boot boot order it's a grub file or it's your device driver it's a file it's a uh libraries it's a file to manage all these files you can

01:02:47 use multiple folder hierarchy so your VM becomes a folder you can in fact check in your when you create a VM in our host operating system you will find that corresponding folder has been created but in case of ram ram is always dynamically allocated for hard disk you have option to choose whether to allocate dynamically or statically but in case of ram it is always dynamically allocated I think answer is obvious why why Ram is always dynamically allocated what if we statically allocate Ram what will

01:03:46 happen sorry come on yes so your uh degree of parallelism will go down because you are pre-allocating the RAM and RAM is a critical resource it's an expensive resource you know for hard disk you can buy one TB hard disk with your pocket money but you cannot buy one TB of main memory with your pocket money of course if you are Amani's kids your pocket money you can buy one TV R but I'm assuming all of you are middle class so R is an expensive resource we don't want to allocate it statistically right

01:04:35 statist statistically so dynamism will give more chance to run more number of applications and using the resources efficiently an operating system is all about scheduling your jobs your programs in a way that maximum utilization of resources can be done can be created any doubt any question here good evening sir yes actually uh time to submit the assignment is 9th February so uh I have a discuss this3 after class yes sir any technical question regarding the lecture which I have covered today or in previous

01:05:30 Le sir uh so as you mentioned for Ram apart from expensive thing like is any other technical reason to be dynamic come again I mean as you mentioned for Ram it as it is expensive we can't afford for static so is there any other reason technically or only that that s I'm asking that's the only reason okay because see uh why you want to dynamically allocate tram is those who require at that particular point of time they should be able to get the Ram otherwise you keep the ram free so that other processes have chance to enter

01:06:11 into memory and execute themselves so uh physical classroom say we occupy this physical classroom for 24 hours other classes won't be able to use this space but our classes are only running from 6 to 7:30 and Tuesday so you only need to occupy this space during 6:00 to 7:30 Tuesday but as a ram if you want to statistically statically allocated we are saying that let's block this classroom for this course only for all the time which is which is not a good idea the classroom is an expensive resource it's a shared resource and if

01:07:06 we are not using it actively someone else should be using it able to use it that's the only reason why we are dynamically allocating registers CPUs Ram or network bwith because they are expensive resources for hard disk over the time they have become cheaper so we started giving option of dynamically or statically allocating them uh any other question online uh there was question related to GPU what about GPU whether they are dynamically allocated or not yes sir then that question I going to ask you that GPU we can use GPU now

01:07:52 also which is uh which can can be taken uh it is expensive but the thing is that we we can dynamically allocate the GPU okay so uh uh those who are using shared GP resources like dgx or a00 machines uh I don't know how you are using are you using through some orchestration or you are directly getting containers we directly getting contain containers so that should not be the efficient way of using shared pool of GPU resources so kuber kuber kubernetes or similar tools who provide orchestration they are good at

01:08:46 scheduling containers based on their uses and requirements so they are kind of job scheduling algorithms for containers but uh I think cber is require some complexity to you know manage and that's why our technical staff is directly allocating some containers to you of fixed size and if you are running that container whether you are running in a program inside that container or not or GPU is occupied for that container which is kind of we are unable to use our resources efficiently so this also you can take up

01:09:33 as a project of this course create an oration platform to manage gpus at FR time another good project when I say good project they are difficult to do is uh RAM allocation is not dynamic in Virtual machines as of or adding a new CPU in your virtual machine configuration is not Dynamic it's not happening at runtime you need to switch off the machine add more resources and then switch on but hard disk you can mount additional hard disk at run time so secondary storage can be Expendable at runtime even in VMS

01:10:27 but uh when you create your virtual machine using virtual box or VM player you get some configur configuration time parameter that how much RAM should be allocate and if you try changing Ram or adding more CPU your virtual box may ask you it's not possible I'm sure about the processor it will not allow to add or remove any allocated processor from VM at runtime Ram I need to check whether they are allowing it now or not but earlier they were not allowing even Ram to expand at run time they were only allowing this to

01:11:15 expand at front time for virtual machines so you can create an extension pack of virtual box or VM player here you manage how to expand RAM or how to add a more new CPU or remove existing Ram or existing CPU from a running VM what all need we need to manage there is PCB that's all PCB program control block with corresponding register values if we are increasing Ram we need to ensure that the now uh RAM is space is available more so your addressing scheme will get changed your existing resour processes which are running on older

01:12:06 addressing scheme they should be updated to new addressing scheme I hope you are getting what I'm saying here with the addressing scheme in your RAM you have this space and you refer to a particular uh flipflop or register in your Ram with some addressing scheme and when you expand it dynamically or reduce it dynamically then your addressing scheme is changing it's like we added new buildings in IJ campus at runtime or removed at run time so addressing scheme where your RO number kind of partitioning uh you can say it's not

01:12:58 about partitioning see uh if you recall from your operating system for your RAM we have a dressing scheme we have that translation look aside TB and based on the memory allocated or the like in Windows XP we discuss about fat 32 it can only have 4GB uh maximum file size because of the addressing scheme in that file system so F your addressing scheme affects a lot how you will reach to a particular PCB or data point or any document library or file or folder so if you're addressing a scheme is changing so dynamically I expanded

01:13:49 this IJ campus and include more area now my addressing scheme will change dynamically my operating system should be able to adapt new addresses at run time but picking this Ram size is a is a boot time parameter so whenever your operating system boots up it checks it runs a uh General checkup on your motherboard that all the hardware are in the space specified configuration and if you open your laptop when I say open means physically open your laptop with the screwdrivers and all you take out Ram slot at the

01:14:40 boot time your operating system will detect that now the ram is not available whatever Ram is only available it will vot with that amount of L you open your laptop you put a new Ram slot new Ram in a available slot and you restart your operating system your RAM will be increased similarly with processor also if it's a compatible architecture you can change the processor but that's a boot time level you cannot run your operating system and while it's running you cannot change the amount of frame or change the

01:15:26 processors you have to switch it off uh shut down your operating system right so those are boot time parameters which you need to manage to handle at runtime if you wish to dynamically manage to add or remove new Ram or add or remove processors and that capability will take you closer to live migration of virtual machines so there are two types of migration possible in live migration itself two types of migrations are possible one live migration may ask you to uh take one VM from one physical machine and take it to another physical

01:16:24 machine and another live migration is changing the configuration of VM itself whether you keep it on the same physical machine or another physical machine doesn't matter but change the configuration at run time like add more RAM remove more RAM remove some Ram or add more processor remove some processor so both are kind of live migration and live migration happens when you do it at run time but when you do it at when you switch it off and then do the migration then you transfer VM to another physical machine

01:17:04 or change the configuration that's the not real time or not live migration and you can easily imagine live migration can be crucial in many application domains you don't want to switch off your machine and then move it to high demand High specification machine or lower specification machine you remember the uh definition of cloud dynamically able to scale up or scale down elastic you know on demand provisioning of resources on demand provisioning of resources means removing resources as well as adding

01:17:52 resources now if on demand I want want to add more resources technically what it requires adding more resources to VM or creating more VMS in the pool if I my system is already running on a distributed kind of setup if my system is already using say 5 PMs and I want to n dynamically add new VM then it should be allowed which is relatively easy but if I wish to enhance the existing 4 VMS only then this live migration will become an issue or become a limitation for me at this point of time so some virtualization companies

01:18:43 promises live migration how does they do it is uh I'll give you analogy and we'll discuss that analogy in the next lecture remind me um did I compare client server architecture with P2P in this class no okay I'll discuss it again I want you to create a pure P2P architecture I think I discussed in this class discussed right or not yes or no no okay I I'll uh I'll discuss you remind me it's your responsibility next Tuesday we'll compare this live migration and we'll see how it is uh you know uh getting the analogy from P2P and

01:19:47 client server architecture they are promising live migration the way whats is promising P2P architecture okay we'll compare this in the next next lecture on Tuesday uh this Saturday there's a ignas event right so regular people have option to either enjoy ignas or enjoy my classes we'll have online class from 2 to 3:30 recording will be available and that will focus on handson about gcp or some other platform which can help you in solving your assignment when sir it will be Saturday this Saturday 2 to 3:30 yes

01:20:43 okay so if you have any question you can stay back otherwise you can leave class is over

Summary

The lecture discussed OS-level and hardware-level virtualization, focusing on performance efficiency, resource management, and the impact of system calls on virtualization.

Highlights

📈 OS-level virtualization shares the kernel, causing performance overhead.

⚙️ Hardware virtualization uses separate kernels, enhancing performance.

🔄 Context switching is more complex in OS-level virtualization.

🛠️ Hardware virtualization allows direct interaction with hardware, reducing latency.

📊 Resource allocation is dynamically managed in virtualization environments.

💻 Live migration in virtualization can enhance resource scaling without downtime.

🔍 The importance of efficient resource management in operating systems was emphasized.

Key Insights

🌐 Performance Comparison: OS-level virtualization has performance overhead due to shared kernel usage, while hardware virtualization benefits from isolated kernel environments, allowing better resource management.

🔒 Resource Management: Dynamic allocation of resources, especially RAM, is crucial for efficient virtualization and prevents resource bottlenecks during peak loads.

⚡ Impact of System Calls: Excessive layers in OS-level virtualization can lead to increased system calls, causing delays in executing applications compared to hardware virtualization.

🔥 Context Switching Challenges: More context switching occurs in OS-level virtualization due to multiple operating systems interacting with a single kernel, impacting overall performance.

🚀 Live Migration: The ability to dynamically allocate resources and migrate virtual machines enhances cloud computing efficiency, allowing for on-demand scaling without downtime.

🎛️ Layered Architecture: The architectural differences between OS-level and hardware virtualization highlight the trade-offs in performance and resource utilization, which are critical for system design decisions.

📚 Educational Focus: The lecture emphasized practical hands-on learning in upcoming classes, reinforcing theoretical concepts with real-world applications.