Homework 2: CS 524

1. **Complete reading Chapter 3 of the textbook and the lecture materials. Please note the errata: The references to [19] on p. 56 of the book should be replaced with references to [20]! Please also read [20] (available free) at https://www.kernel.org/doc/ols/2007/ols2007v2-pages-87-96.pdf.**

ANS.

Completed reading Chapter 3 from the Textbook

2. **Explain the advantage that paravirtualization provides for handling timers in virtual machines.**

ANS.

All modern operating systems, even idle virtual machine rely on clock interrupts to maintain their internal timers, a feature that is particularly essential for real-time media processing. With paravirtualization, the virtual machine code is changed to request a notification at the specified time; without it, the hypervisor would need to schedule timer interrupts for idle machines when the guest operating system is scheduled back to run, which wouldn't be a scalable way to implement virtualization.

(References: Textbook: Cloud Computing: Business Trends and Technologies)

3. **Explain how paravirtualization helps in minimizing access to APIC.**

ANS.

In conventional manner, an operating system deals with multiple CPUs in the same way it deals with one; with modular design, it is just the scheduler and the interrupt handlers that need to be fully aware of the differences. Without delving into the depths here, we note that x86-based multi-processor architectures use the Advanced Programmable Interrupt Controller (APIC) for interrupt redirection in support of Symmetric Multi-Processing (SMP). Accessing APIC in virtual mode is expensive because of the transitions into and out of the hypervisor. With paravirtualization, which has the full view of the code, the multiple APIC access requests can be replaced with a single hypercall.

(References: Textbook: Cloud Computing: Business Trends and Technologies)

4. **Find out if *Linux* (like *Unix*) has both the user-mode and system-mode stacks for each process it runs.**

ANS.

Yes, Linux has both the user-mode and system-mode stacks for each process it runs. All modern CPUs have more than one set of identical registers. As a minimum, one register set is reserved for the user mode—in which application programs execute– and the other for the system (or supervisory, or kernel) mode, in which only the operating system software executes. The reason for this is Interrupt Handling. The first thing that happens at interrupt or exception is that the CPU switches to the system mode, which automatically activates the system stack pointer. This would allow it to save the CPU state and once the system mode requests to switch back, the state is restored and goes back to the user mode.

(References: Textbook: Cloud Computing: Business Trends and Technologies)

5. Find out what "unscrambled" means in the description of the *Intel LSL* instruction (you can, for example, use the Intel manual referenced in the lecture).

ANS.

According to the Intel manual, "unscrambled" in Intel Load Segment Limit (LSL) instruction means the limit scaled according to the setting of the G flag in the segment descriptor. It loads the unscrambled segment limit from the segment descriptor specified with the second operand (source operand) into the first operand (destination operand) and sets the ZF flag in the EFLAGS register. If the segment descriptor cannot be accessed or is an invalid type for the instruction, the ZF flag is cleared and no value is loaded in the destination operand.

To make sure that the offset is within the limit of the segment when the processor accesses any segment it performs a limit check. The software can perform the limit checking using the LSL (Load Segment Limit) instruction. The LSL instruction specifies the segment selector for the segment descriptor whose limit is to be checked and a destination register. Depending on the G flag. The limits are interpreted in a different way. The segment limit is a 20-bit value contained in bytes 0 and 1 and in the first 4 bits of byte 6 of the segment descriptor. When the G flag is clear, the effective limit is the value of the 20-bit limit in the segment descriptor. Here, the limit ranges from 0 to 1MB when the G flag is set 4KB page granularity, the processor scales the value in the limit field by a factor of 212 (4KBytes). In this case, the effective limit ranges from 4KB to 4GB.

(References: Intel® 64 and IA-32 Architectures Software Developer's Manual Volume 2 (2A, 2B, 2C & 2D): Instruction Set Reference, A-Z)

6. Read the following two papers:

- **Carl Waldspurger and Rosenblum, M. (2012) *I/O Virtualization.* Communications of the ACM, vol. 55, No 1. January 2012. Pages 66-72; and**

- **Muli Ben-Yehuda; Xenidis, J.; Ostrowski, M.; Rister, K.; Bruemmer, A.; Van Doorn, L. (2007). *The Price of Safety: Evaluating IOMMU Performance.* Proceedings of the Linux Symposium on June 27th–30th, 2007. Ottawa, Ontario. Pages 225-230.**

a) **Explain the advantages and disadvantages of using I/O MMU by citing the appropriate text from the paper.**
b) **Research the Web to find what is meant by "carrier-grade hypervisors". What products are available?**

ANS.

a) The various advantages and disadvantages of using I/O MMU are:

ADVANTAGES:

- Decoupling enables time and space multiplexing of I/O devices, allowing multiple logical devices to be implemented by a smaller number of physical devices.
- The ability to multiplex logical I/O devices onto physical ones allows both administrators and automated systems to drive I/O devices at higher utilization and achieve better hardware efficiency. Much about virtualization rapid adoption over the past decade can be attributed to the significant cost savings resulting from such basic partitioning and server consolidation.
- Decoupling provides for flexible mappings between logical and physical devices, facilitating seamless portability. By supporting mappings of logical I/O devices to physical devices with different yet semantically compatible interfaces, virtualization makes VMs portable, even across heterogeneous systems.
- Decoupling also enables popular VM features such as the ability to suspend and resume a VM and the ability to move a running VM between physical machines, known as live migration.
- This virtualization layer may also change mappings to physical devices, even when the VM itself does not move. For example, by changing mappings while copying, storage contents, a VM's virtual disk can be migrated transparently between network storage units, even while remaining in active use by the VM.
- The same capability can be used to improve availability or balance load across different I/O channels.
- I/O virtualization provides a foothold for many innovative and beneficial enhancements of the logical I/O devices.
- One useful capability enabled by I/O virtualization is device aggregation, where multiple physical devices can be combined into a single more capable logical device that is exported to the VM. Examples include combining multiple disk storage devices exported as a single larger disk.
- New features can be added to existing systems by interposing and transforming virtual I/O requests, transparently enhancing unmodified software with new capabilities. For example,

a disk write can be transformed into replicated writes to multiple disks, so that the system can tolerate disk-device failures.

- Many I/O virtualization enhancements are designed to improve system security.
- Memory is also protected from malicious devices that are attempting DMA attacks and faulty devices that are attempting errant memory transfers because a device cannot read or write to memory that has not been explicitly allocated (mapped) for it. The memory protection is since OS running on the CPU exclusively controls both the MMU and the IOMMU. The devices are physically unable to circumvent or corrupt configured memory management tables.
- I/O MMU translates the I/O virtual memory address to corresponding physical memory, making direct access by devices safe and efficient and allows the driver in the VM to program device DMA using its virtualized notion of memory address, while allowing the hypervisor to decide where VM memory is located.
- The large region of memory can be allocated without the need to be contiguous in physical memory – the IOMMU maps contiguous virtual addresses to the underlying fragmented physical addresses.
- Devices that do not support memory addresses long enough to address the entire physical memory can still address the entire memory through the IOMMU, avoiding overheads associated with copying buffers to and from the peripheral's addressable memory space.

DISADVANTAGES:

- Some degradation of performance of translation and management overhead (e.g., page table walks).
- Consumption of physical memory for the added I/O page (translation) tables. This can be mitigated if the tables can be shared with the processor.

(References: I/O Virtualization | January 2012 | Communications of the ACM, Input–output memory management unit - Wikipedia)

b) "Carrier Grade Virtualization" (CGV) is a strategic software component whose integration into carrier grade platforms must preserve existing carrier grade performance and attributes. CGV targets more than core network equipment and can be applied to a much wider range of equipment types (network appliances, firewalls, gateways, mid-tier routers, PBXs, office equipment, etc.) and lets OEMs realize benefits of higher availability in an evolutionary fashion, without need for revolutionary re-architecture of hardware and software platforms.

Carrier Grade can be defined as virtualization services that fulfil some or all expected properties existing in carrier grade solution. Carrier Grade Virtualization reduces the cost and complexity of maintaining carrier grad properties in edge and core network elements such as IP Multimedia Systems (IMS) nodes. Also networking and telecommunication OEMs can reuse existing investments in their carrier grade system while gaining the benefits of using real-time virtualization software.

Some products available are Bare-metal Xen Hypervisor, Oracle Solaris etc.

Some of its characteristics are:

1. Availability
2. High performance scaling
3. Small error recovery domains
4. Real-time behavior
5. Upgrade capabilities
6. Configurable security
7. Efficient and Uniform management services

(References: http://www.linuxpundit.com/documents/CGV_WP_Final_FN.pdf)

7. Find out what hypervisors *Amazon* is using in EC2 and describe their major characteristics.

ANS.

EC2 uses two different hypervisors, depending on the generation of instance that you are using. The earlier generation instances used a modified version of the Xen hypervisor, and the newer generation uses the Nitro hypervisor, which was developed by AWS themselves, and is based on a customized version of KVM.

The differences are as below:

PARAMETER	XEN	KVM
Virtualization	Full hardware virtualization + Para virtualization	Full hardware virtualization
Age	Older and mature	Comparatively new
Host OS	Linux, UNIX	Linux
Stability	Excellent flexibility and performance	Offer flexibility and performance but not as much as XEN
Performance	Dedicated resources thus no performance issues	Issues in performance
networkinterview.com (An Initiative By ipwithease.com)		

Various features are:

a. Advanced Memory Management
b. Resource Management
c. Scalability
d. High Availability and Fault Tolerance
e. Network and Storage
f. Security

Major characteristics for KVM are:

Live VM Migration: It supports virtual machine live migration from one host to another allows workload balancing and the avoidance of downtime.

Live Storage Migration: Move live running virtual machines and their associated virtual disk image within and across resource pools leveraging local and shared storage.

Host Failure Protection: Deliver high availability by automatically restarting virtual machines if a failure occurs at the VM, hypervisor or server level. Link aggregation bonds network interfaces for network redundancy and increased throughput.

Host Power Protection: Take advantage of embedded hardware features to lower datacenter electricity consumption by dynamically consolidating VMs on fewer systems and then powering off underutilized servers as demand for services fluctuates.

Memory Overcommit: Reduce costs and improve application performance and protection by sharing unused server memory between VMs on the host server.

Site Recovery: Provides site-to-site disaster recovery planning and services for virtual environments. Site recovery is easy to set up, fast to recover, and can frequently test to ensure disaster recovery plans remain valid.

Major characteristics for KVM are:

- hot plug vCPUs
- Dynamic memory management
- Live Migration
- Memory write intensive workload impacts in migration process

(References: https://wiki.xenproject.org/wiki/Xen_Project_Release_Features#Features, https://www.quora.com/What-is-the-underlying-hypervisor-for-EC2, https://en.wikipedia.org/wiki/Amazon_Elastic_Compute_Cloud, https://en.wikipedia.org/wiki/Kernel-based_Virtual_Machine)

8. Examine the *Amazon* EC2 VM offer capabilities and particularly the Amazon Machine Image (AMI) ((https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/AMIs.html (Links to an external site.)) and answer the following questions:

1. How (i.e., in what units) does EC2 measure the CPU power of a virtual machine and how is the unit in question translated into the power of the physical processors)?

2. **What kinds of machine instances are there as characterized by the power of their respective CPUs, platform (i.e., 32-bit or 64-bit), memory, storage, etc.? Please list all the instances in the nomenclature along with their respective characteristics.**
3. **Which operating systems are available on the above systems?**
4. **What is an AMI and what is its relationship to an *instance*?**
5. **What are the components of an AMI?**

ANS.

1. The amount of CPU that is allocated to a particular instance is expressed in terms of these EC2 Compute Units. We use several benchmarks and tests to manage the consistency and predictability of the performance from an EC2 Compute Unit. The EC2 Compute Unit (ECU) provides the relative measure of the integer processing power of an Amazon EC2 instance.EC2 measures the CPU power of a virtual machine in ECU (EC2 Computing Units).

   ECU equates to a certain amount of computing cycles in a way that is purportedly independent of the actual hardware. So, several benchmarks and tests are used to determine how the Computing Units translate into power of the physical processor. According to the documentations, a single ECU is defined as the compute power of a 1.0 to 1.2 GHz of a 2007 server CPU capacity.

   As an example, the most common and oldest instance type, m1.large, is rated at 4 ECUs (2 cores of 2 ECUs each).

   (References: Are All AWS ECUs Created Equal? | Datadog (datadoghq.com), Amazon EC2 FAQs - Amazon Web Services)

2. Amazon EC2 gives you the option of choosing between 10 different instance types, distributed across 6 instance families. You have the flexibility to choose the combination of instance types and sizes most appropriate for your application today, and you can always change the type you use later as your business and application needs change.

   The various Instance Features are:
   - Burstable Performance Instances
   - Multiple Storage Options
   - EBS-optimized Instances
   - Cluster Networking
   - Intel Processor Features like AVX, Turbo Boost, DL Boost etc

   The available instance families and instance types are:

   General-Purpose.

   This family includes the M1 and M3 instance types, both of which provide a balance of CPU, memory, and network resources making them a good choice for many applications. For many of you, this family is often the first choice, with sizes ranging

from 1 vCPU with 2 GiB of RAM to 8 vCPUs with 30 GiB of RAM. The balance of resources makes them ideal for running small and mid-size databases, more memory-hungry data processing tasks, caching fleets, and backend servers for SAP, Microsoft SharePoint, and other enterprise applications.

M3 instances are the newest generation of general-purpose instances and give you the option of a larger number of virtual CPUs (vCPUs) that provide higher performance. M3 instances are recommended if you are seeking general-purpose instances with demanding CPU requirements. M1 instances are the original family of general-purpose instances and provide the lowest cost options for running your applications. M1 instances are a great option if you want smaller instance sizes with moderate CPU performance, and a lower overall price.

Compute-Optimized.

This family includes the C1 and CC2 instance types and is geared towards applications that benefit from high compute power.

Compute-optimized instances have a higher ratio of vCPUs to memory than other families and the lowest cost per vCPU of all the Amazon EC2 instance types. If you are running any CPU-bound scale-out applications, you should look at compute-optimized instances first. Examples of such applications include front end fleets for high-traffic web sites, on-demand batch processing, distributed analytics, web servers, video encoding, and high performance science and engineering applications like genome analysis, high-energy physics, or computational fluid dynamics.

CC2 instances are the latest generation of compute-optimized instances and provide the lowest cost for CPU performance for all Amazon EC2 instance types. In addition, CC2 instances provide a number of advanced capabilities: Intel Xeon E5-2670 processors; high core count (32 vCPUs); and support for cluster networking. These capabilities allowed us to create a cluster of 1064 CC2 instances that achieved a Linpack score of 240.09 Teraflops, good for an entry at number 42 in the November 2011 Top500 supercomputer list.

C1 instances are the first generation of compute-optimized instances. They are available in smaller sizes and are ideal for massively scaled-out applications at massive scale. Most examples of customers launching 1000s of instances to transcode videos or for virtual drug design are likely to take advantage of C1 instances.

Memory-Optimized.

This family includes the M2 and CR1 instance types and is designed for memory-intensive applications. Instances in this family have the lowest cost per GiB of RAM of all Amazon EC2 instance types. If your application is memory-bound, you should use these instances. Examples include high performance databases and distributed cache, in-

memory analytics, genome assembly, and larger deployments of SAP, Microsoft SharePoint, and other enterprise applications. In general, if you are running a performance-sensitive database you should first look at this family.

CR1 instances are the latest generation of memory-optimized instances and provide more memory (244 GiB), faster CPU (Intel Xeon E5-2670) compared to M2 instances. CR1 instances also support cluster networking for bandwidth intensive applications.

M2 instances are available in smaller sizes and are an excellent option for many memory-bound applications.

Storage-Optimized.

This family includes the HI1 and HS1 instance types and provides you with direct-attached storage options optimized for applications with specific disk I/O and storage capacity requirements. Currently there are two types of storage-optimized instances.

HI1 instances are optimized for very high random I/O performance and low cost per IOPS. These instances can deliver over 120,000 4k random read IOPS making them ideal for transactional applications. In particular, we designed these instances to be the best platform for large deployments of NoSQL databases like Cassandra and MongoDB.

HS1 instances are optimized for very high storage density, low storage cost, and high sequential I/O performance. HS1 instances give 48 TB of storage capacity across 24 hard disk drives, high network performance, and are capable of supporting throughput performance of as much as 2.6 GBps. These instances are designed for large-scale data warehouses, large always-on Hadoop clusters, and for cluster file systems. Indeed, HS1 instances are the underlying instance type for our petabyte-scale data warehousing service, Amazon Redshift.

Micro Instances.

Micro, or T1, instances are a very low-cost instance option providing a small amount of CPU resources. Micro instances may opportunistically increase CPU capacity in short bursts when additional cycles are available. They are well suited for lower throughput applications like bastion hosts or administrative applications, or for low-traffic websites that require additional compute cycles from time to time.

Micro instances are available in the AWS Free Usage Tier to allow you to explore EC2 functionality at no charge. Due to the opportunistic scheduling used by Micro instances, you should not use them for applications that require sustained CPU performance. You can learn more about the characteristics of Micro instances and appropriate workload characteristics in the Amazon EC2 documentation.

GPU Instances.

This family includes the CG1 instance type and allows you to take advantage of the parallel performance of NVidia Tesla GPUs using the CUDA or OpenCL programming models for GPGPU computing. GPU instances also provide high CPU capabilities and support cluster networking. For applications like AMBER, a molecular dynamics application, you can get 4-5x improvement in performance compared to CC2 instances. Many of you are running computational chemistry, rendering, and financial analysis applications on CG1 instances today to take advantage of the speedup you can get from GPGUs.

(References: Amazon EC2 Instance Types - Amazon Web Services, Choosing the Right EC2 Instance Type for Your Application | AWS News Blog (amazon.com))

3. The operating systems currently available to use with Amazon EC2 instances include:
   - Amazon Linux
   - Windows Server 2012
   - CentOS 6.5
   - Debian 7.4

   (References: Amazon EC2 Features - Amazon Web Services)

4. An Amazon Machine Image (AMI) provides the information required to launch an instance. You must specify an AMI when you launch an instance. You can launch multiple instances from a single AMI when you need multiple instances with the same configuration. You can use different AMIs to launch instances when you need instances with different configurations.

   (References: https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/AMIs.html)

5. An Amazon Machine Image (AMI) includes the following components:

   - One or more Amazon Elastic Block Store (Amazon EBS) snapshots, or, for instance-store-backed AMIs, a template for the root volume of the instance (for example, an operating system, an application server, and applications).
   - Launch permissions that control which AWS accounts can use the AMI to launch instances.
   - A block device mapping that specifies the volumes to attach to the instance when it's launched.

   (References: https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/AMIs.html)

9. Find out about the pricing of the EC2 platforms and provide a few examples

ANS.

Amazon EC2 is free to try. You can also pay for Dedicated Hosts which provide you with EC2 instance capacity on physical servers dedicated for your use. AWS Free Tier includes 750 hours of Linux and Windows t2.micro instances, (t3.micro for the regions in which t2.micro is unavailable) each month for one year. To stay within the Free Tier, use only EC2 Micro instances.

There are five ways to pay for Amazon EC2 instances: On-Demand, Savings Plans, Reserved Instances, and Spot Instances.

On-Demand Instance
With On-Demand instances, you pay for compute capacity by the hour or the second depending on which instances you run. No longer-term commitments or upfront payments are needed. You can increase or decrease your compute capacity depending on the demands of your application and only pay the specified per hourly rates for the instance you use.

Spot Instances
Amazon EC2 Spot instances allow you to request spare Amazon EC2 computing capacity for up to 90% off the On-Demand price.

Savings Plans
Savings Plans are a flexible pricing model that offer low prices on EC2 and Fargate usage, in exchange for a commitment to a consistent amount of usage (measured in $/hour) for a 1- or 3-year term.

Reserved Instances
Reserved Instances provide you with a significant discount (up to 75%) compared to On-Demand instance pricing. In addition, when Reserved Instances are assigned to a specific Availability Zone, they provide a capacity reservation, giving the additional confidence in the ability to launch instances when needed.

Dedicated Hosts
A Dedicated Host is a physical EC2 server dedicated for your use. Dedicated Hosts can help you reduce costs by allowing you to use your existing server-bound software licenses, including Windows Server, SQL Server, and SUSE Linux Enterprise Server (subject to your license terms), and can also help you meet compliance requirements.

Below are the prices for On-Demand instances:

Region: US East (Verizon) - New York

	vCPU	ECU	Memory (GiB)	Instance Storage (GB)	Linux/UNIX Usage
General Purpose - Current Generation					
t3.medium	2	Variable	4 GiB	EBS Only	$0.056 per Hour
t3.xlarge	4	Variable	16 GiB	EBS Only	$0.224 per Hour
GPU Instances - Current Generation					
g4dn.2xlarge	8	N/A	32 GiB	225 GB NVMe SSD	$1.317 per Hour
Memory Optimized - Current Generation					
r5.2xlarge	8	37	64 GiB	EBS Only	$0.68 per Hour

(References: Amazon EC2 Pricing - Amazon Web Services, EC2 On-Demand Instance Pricing – Amazon Web Services)

10. From the above exercise, you will learn that it is possible to create a free machine instance. Please, do the following:

1. **Find out and document the essence of the respective *Service Level Agreement (SLA)* on; in particular write down what one needs to do in order to maintain this service free.**
2. **Describe the process (i.e., what exactly one needs to do) to create a free machine instance that could be used as a server. (Do not, however, create anything yet!)**
3. **Can you create a machine instance equivalent to your own PC and then transfer your own PC image there? If so, how would you achieve that?**

ANS.

1. A **Service Level Agreement (SLA)** is the bond for performance negotiated between the cloud services provider and the client. Earlier, in cloud computing all Service Level Agreements were negotiated between a client and the service consumer. Service Level Agreements usually specify **some parameters** which are mentioned below:
    2. Availability of the Service (uptime)
    3. Latency or the response time
    4. Service components reliability
    5. Each party accountability
    6. Warranties

    In any case, if a cloud service provider fails to meet the stated targets of minimums then the provider has to pay the penalty to the cloud service consumer as per the agreement.

The AWS Free Tier makes certain amounts and types of resources for new AWS accounts available free of charge for a one-year period. Any amounts and types of resources that aren't covered are charged at standard rates.

To avoid unnecessary charges:

- Understand what services and resources are covered by the AWS Free Tier.
- Monitor Free Tier usage with AWS Budgets.
- Monitor costs in the Billing and Cost Management console.
- Be sure that your planned configuration falls under the Free Tier offering.
- Clean up test resources when you're done using them.

(References: Service level agreements in Cloud computing - GeeksforGeeks, Prevent Free Tier Charges (amazon.com))

2. The process what exactly one need to do to create a free machine instance, that could be used as a server are followed:

   a. Login to your amazon account
   b. Create an instance
   c. Choose AMI
   d. Configure the instance as database instance and select storage needs
   e. Configure it with the web applications and customize various factors according to the needs.
   f. Deploy our instance

   (References: https://www.guru99.com/creating-amazon-ec2-instance.html, Amazon Machine Images (AMI) - Amazon Elastic Compute Cloud)

3. Yes, we can create a machine instance that is identical to my own PC and then upload our own PC image to it. All of this can be achieved by setting up an Amazon EC2 instance and hosting it as a server. After that, we must connect our computer to the server and migrate the image.

   VM Import/Export enables you to easily import virtual machine images from your existing environment to Amazon EC2 instances and export them back to your on-premises environment. This offering allows you to leverage your existing investments in the virtual machines that you have built to meet your IT security, configuration management, and compliance requirements by bringing those virtual machines into Amazon EC2 as ready-to-use instances. You can also export imported instances back to your on-premises virtualization infrastructure, allowing you to deploy workloads across your IT infrastructure.

To import your images, use the AWS CLI or other developer tools to import a virtual machine (VM) image from your VMware environment. If you use the VMware vSphere virtualization platform, you can also use the AWS Management Portal for vCenter to import your VM. As part of the import process, VM Import will convert your VM into an Amazon EC2 AMI, which you can use to run Amazon EC2 instances. Once your VM has been imported, you can take advantage of Amazon's elasticity, scalability and monitoring via offerings like Auto Scaling, Elastic Load Balancing and CloudWatch to support your imported images.

(References: https://aws.amazon.com/ec2/vm-import/)