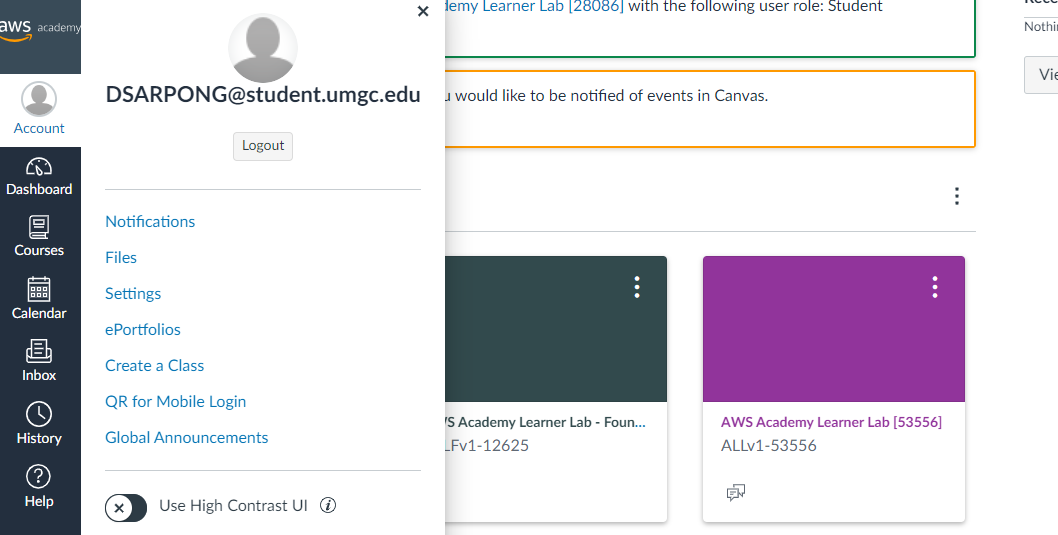
**CMIT 495 Current Trends and Projects in Computer Networks and Security**

*Week 1 – Virtualization*

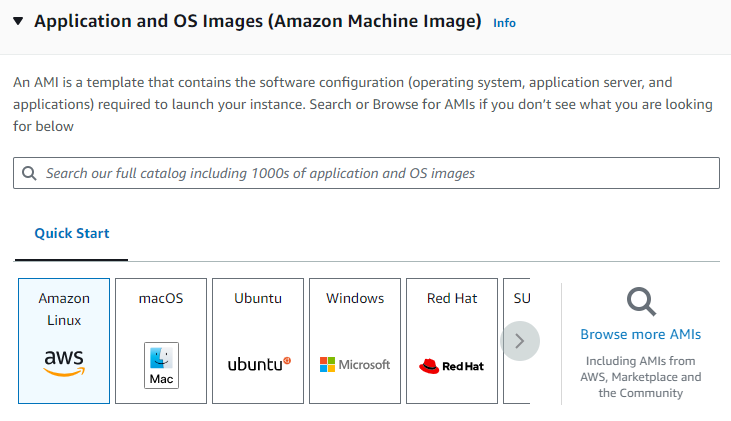
1. **Log in to your newly created AWS account and take a screen capture of the AWS Console (Dashboard) and embed it below.**



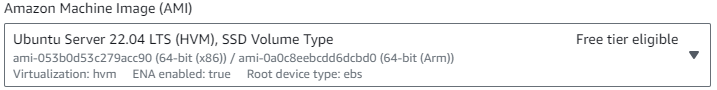
email

1. **Provide a detailed overview of the steps required to provision (create) and launch an AWS Ubuntu-based server virtual instance. The steps may be listed in bullet points or complete sentences. Use as much space as required.**

After launching the AWS learner's lab click to start lab. Next click AWS which is located on top left of the screen. After AWS opens, go to services and click EC2, under this, click instances and then launch instances.

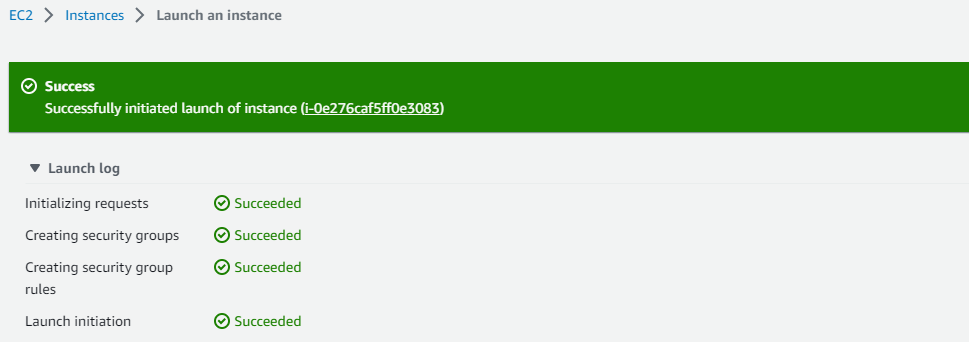


Under application and OS images, select Ubuntu. Here select Ubuntu Server 22.04 LTS (HVM) SSD Volume Type, 64 bit (x86). Under instance type, select t2.micro under free tier.



Under key pair, select create a new key pair. Name it 'Key 1', the type should be RSA, set the private key to be of .pem extension. Next click create key.

After this the key will be downloaded to your machine, scroll down and on the right bottom of the screen, click launch instance.



1. **What are the benefits of virtualization in a cloud environment? Discuss a minimum of three benefits in detail.**

a. Migration and Portability: It is simpler to move workloads between several physical servers or even between cloud providers thanks to virtualization, which creates an abstraction layer between the operating system and the underlying hardware. In addition to increasing vendor flexibility, its portability prevents vendor lock-in. Applications may be transferred to many settings with little to no modification, lowering the risks involved with switching infrastructure or cloud providers.

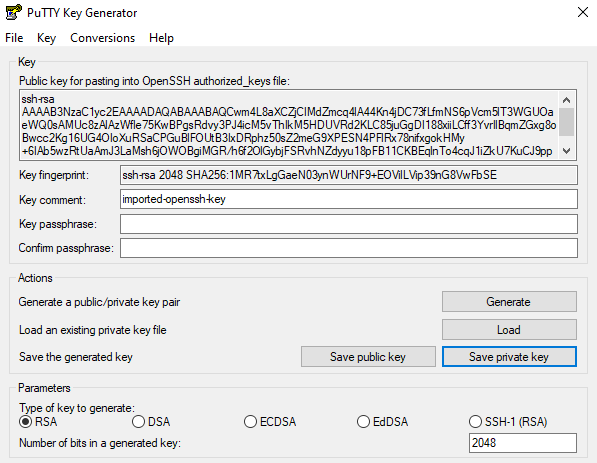
b. Cost Reduction: Cost Savings: By maximizing resource consumption and allowing numerous workloads to run on a single server, virtualization lowers hardware expenses. As fewer physical servers are needed as a result, capital expenses are reduced. Additionally, virtualization makes it possible to manage resources effectively, which lowers operational expenses for things like electricity, cooling, and physical space. Organizations can increase usage levels without compromising performance or dependability, which improves return on investment.

c. Flexibility and Scalability: Flexibility and Scalability: By enabling swift provisioning and deprovisioning of VMs, virtualization makes it simpler to scale up or down in response to workload demands. This adaptability is especially helpful in a cloud setting where it's critical to be able to react quickly to shifting requirements. It takes less time to launch new services thanks to virtualization, which allows IT professionals to quickly spin up new instances, distribute resources, and install applications. By letting developers to construct and test apps in separate VMs without affecting the production environment, this also facilitates testing and development environments.

1. **Based on your experience, what was the most challenging aspect of provisioning and launching the AWS Ubuntu-based server instance in the AWS virtual environment?**
2. **Configure your local host computer to use an SSH client application to interact with the newly created and running AWS Ubuntu virtual instance. If using a Windows-based local computer, read over the AWS document, "Connect to your Linux instance from Windows using PuTTY”, download and install PuTTY[[1]](#footnote-1) or use Windows 10 built-in OpenSSH client.**

Installing and downloading PuTTY is the first step. The version includes a puTTYgen utility that converts the private keys we downloaded when we created the instance to a puTTY-acceptable extension.

By selecting it from the start button, we first run the PuTTYgen.

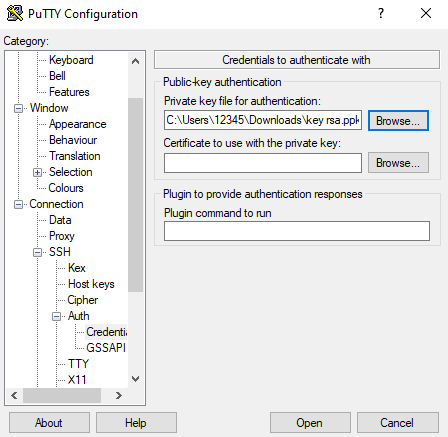


Next, we select the option to "launch an existing private key." Here, we upload the pem-extended key that was previously established. The newly converted key is then saved. When asked if we need to use a passphrase, we respond that we are certain we don't. The new key's extension is ppk.

1. **From the local computer establish an SSH connection to the AWS Ubuntu Server virtual instance, login, and update the Ubuntu Server using *sudo apt-get update* and *sudo apt-get upgrade*. Perform a screen capture after having each command successfully run.**

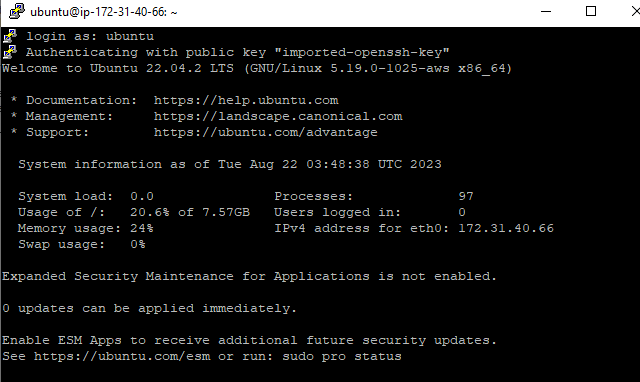
- After that, we run the puTTY application and enter the freshly generated key. The hostname, which is made up of the IP address and the DNS address, must first be entered. The hostname in my case is ec2-107-20-121-161.compute-1.amazonaws.com. The connection type is SSH, and the port number is 22.

Under category, expand SSH, expand Auth then select credentials. Under private key file authentication, browse to upload the private key generated through the Putty generator. Then click open. Click accept on next stage.



You will be required to specify whom you are logging in as.

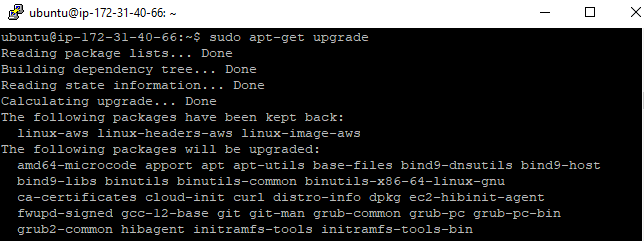




Update:



Upgrade:

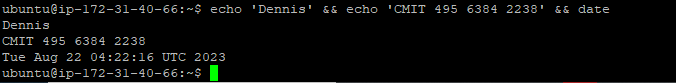


1. **Next, describe in depth what the *update* and *upgrade* commands are doing (explain why it is important to run these commands, how often should these commands be run, where do the commands pull the updates).**

To update the system's available packages and versions, the update command must be executed first. These packages are not installed by the program; it only lists them. To install the list of packages and their versions, we must issue the upgrade command. These commands are crucial because they address any problems from earlier versions and, more critically, any security problems. It is advised to execute these commands as frequently as once each week.

1. **After successfully deploying the Ubuntu Server operating system updates and upgrades, reconnect to the AWS EC2 console, and if needed login again via an SSH client (Note: SSH session inactivity timeout is short). Instructions to SSH are provided by Amazon on how to do so. When connected, run the following commands to obtain information about the host and network settings (perform a screen capture of the results and embed below):**

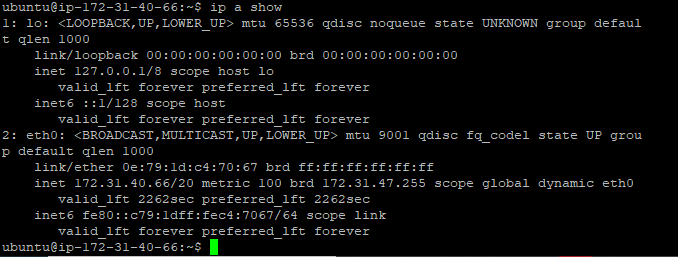
***echo '***Dennis***' && echo 'CMIT 495*** 63842238***' &&*** fall 2023



***whoami***



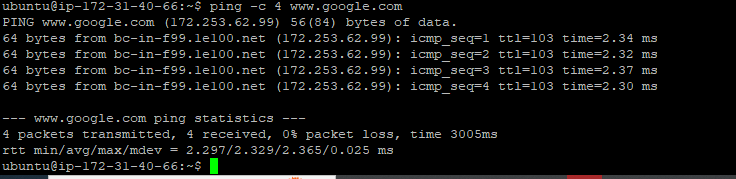
***Ip***



***Pwd***



**ping -c 4** [**www.google.com**](http://www.google.com)



1. **What kind of account did you discover when you ran the *whoami* command? How do you know?**

This gives my current username which is Ubuntu

1. **Note the difference between IP addresses—specifically, when you perform the *ip a show* or *ipconfig* on your personal system, versus the *ip a show* command within the EC2 instance VM command prompt. Describe the network settings of each system. Why are the IP addresses different? Are the IP addresses private or public? What is the difference between a public and private IP address? Explain in detail.**

Because of the nature of the networks they are a part of, the IP addresses you see when running the ip a display command on your personal computer and within an EC2 instance are different. Public IP addresses are used for internet communication, whereas private IP addresses are used for internal communication within closed networks. To ease internal communication within the AWS infrastructure and connectivity with the internet, EC2 instances on AWS have both private and public IP addresses.

EC2 instances in AWS are frequently given both private and public IP addresses:

Private IP Address: Within the Virtual Private Cloud (VPC) in which it sits, each EC2 instance is given a private IP address. As a result, instances inside the same VPC can connect with one another using their personal IP addresses.

Public IP Address: By default, EC2 instances are also given a public IP address, enabling direct internet communication. For operations like SSHing into the instance or gaining access to web services hosted on it, this public IP address is frequently utilized.

1. **Virtualization allows us to place the functionality of many servers into a single host while maintaining configurable levels of separation between all virtual machines. How do you believe virtualization could help in data center consolidation? What would you be leery about during and after any transition?**

Benefits of Virtualization for Consolidating Data Centers:

Resource optimization: Virtualization makes it possible to use hardware resources effectively. You may make the best use possible of the CPU, memory, storage, and network resources by operating numerous virtual machines on a single physical server. As a result, fewer unused physical servers need to be maintained.

Cost Savings: By consolidating servers, you can save money on hardware purchases, energy use, cooling, and physical space. Lower capital and operational costs result from having fewer physical servers.

Energy Efficiency: A data center that houses fewer physical servers uses less energy and requires less cooling, which helps the environment.

Better Scalability: Virtualization makes it simple to provision and scale VMs. You can dynamically allocate resources to VMs as workload demands vary without the need for additional real hardware.

Management is easier when there are fewer physical servers to manage as opposed to many separate servers. Administrative activities can be streamlined with the use of virtualization's centralized management tools.

Disaster recovery planning is made easier and availability is increased with the help of virtualization systems' live migration and snapshot functionality.

worries both during and after the transition:

Impact on performance: Although virtualization strives to optimize resources, sharing physical resources among numerous VMs has the potential to degrade performance if not properly handled. In order to prevent performance bottlenecks, monitoring and efficient resource allocation are essential.

Security risks include the possibility of VM-to-VM attacks when hosting numerous virtual machines on a single physical host. To stop unwanted access and data breaches, isolation and security measures are required.

Management of a virtualized environment can be challenging, particularly as the number of VMs rises. To reduce complexity, appropriate planning and management techniques are required.

Switching to a particular virtualization platform can result in vendor lock-in. Long-term viability depends on ensuring compatibility and flexibility with various virtualization technologies.

Backup and recovery: In order to guarantee data availability and integrity in virtualized settings, proper backup and recovery procedures must be implemented.

Resource management: Although virtualization makes it possible to use resources more effectively, doing so might cause performance problems and resource contention. Planning carefully is necessary to balance resource allocation.

Compliance and Licensing: Virtualization licensing models can be intricate. When consolidating, organizations must make sure that the license conditions are followed and take the cost implications into account.

Single Point of Failure: The underlying physical host can still fail, even while virtualization increases availability thanks to features like live migration. Redundancy and failover measures must be implemented.

1. **Does virtualization increase the cybersecurity posture of the organization? If so, describe how and why. If not, describe how and why not.**

Virtualization can improve an organization's cybersecurity posture, but the degree of the gain relies on a number of variables and how virtualization is used. Let's investigate how virtualization may affect cybersecurity:

1. Segmentation and Isolation: With virtualization, several separate virtual machines (VMs) can be built on a single physical server. Each VM runs separately and has its own set of programs and operating system. This seclusion lessens the effects of security lapses. If only one VM is compromised, the attacker's access may be restricted to that one, lowering the possibility of lateral network movement.

2. Sandboxing: Sandbox environments for testing potentially dangerous software or files can be created using virtualization. Security experts may examine and comprehend risks in these sandboxes without endangering the actual production environment.

3. Snapshot and recovery features: Platforms for virtualization frequently provide snapshot and backup functions. These snapshots record a virtual machine's status at a particular moment in time. Organizations can roll back to a clean snapshot in the case of a security incident or breach, thereby undoing any compromised state.

4. Better Patch administration: By centralizing updates to host systems and virtualized guest systems, virtualization helps simplify patch administration. By doing so, the attack surface can be minimized and the most recent security fixes can be applied to all computers.

5. Intrusion Detection and Prevention: Platforms for virtualization can include capabilities for keeping track of and controlling network traffic between virtual machines and the outside world. Systems for detecting and preventing intrusions (IDS/IPS) can be used to track traffic and stop harmful behavior.

6. Encrypted Communication: To improve data security and privacy, virtual networks can be set up to use encrypted communication channels between virtual computers.

7. Rapid Service Deployment and Isolation: Virtualization makes it possible to quickly roll out new services and applications. This can help businesses quickly create secure testing and development environments, lowering the chance of introducing security flaws into live systems.

8. Disaster Recovery: By enabling businesses to duplicate virtual machines (VMs) and applications across physical servers or even to remote locations, virtualization can simplify disaster recovery. This improves the company's capacity to bounce back from security breaches or natural calamities.

There are nonetheless potential restrictions and factors to take into account:

1. Vulnerabilities in the hypervisor: The hypervisor, which controls the virtualized environment, is a crucial element. All virtual machines (VMs) using that hypervisor may be in danger if the hypervisor itself is compromised.

2. Resource Sharing: Virtualization's ability to isolate users from one another is a plus, but sharing resources on the same physical server might result in resource contention. The performance of other VMs on the same server may be affected if one VM encounters a spike in resource demand as a result of an attack.

3. Complexity: If not effectively managed, virtualized environments can add complexity to the infrastructure, thereby increasing the number of attack vectors.

4. Misconfiguration: Incorrect configuration of virtual environments might result in security flaws. Organizations need to make sure VMs are configured, patched, and secured correctly.

1. **There are various virtualization options: bare-metal (Type 1) in which the hypervisors run directly on the hardware as their own operating systems, and user-space (Type 2) hypervisors that run within the conventional operating systems. Which of these options is more secure? Describe the vulnerabilities you believe exist in either Type 1, Type 2, or both configurations. What do you believe can be done to mitigate these vulnerabilities?**

Hypervisors of Type 1 (Bare-Metal): Hypervisors of Type 1 operate directly on hardware without the aid of a host operating system. Due to their smaller attack surface and improved isolation from the host environment, Type 1 hypervisors are typically regarded as being more secure than Type 2 hypervisors. They are not impervious to weaknesses, though.

Vulnerabilities:

Vulnerabilities in the Type 1 hypervisor itself have the ability to jeopardize the security of all the guest virtual machines (VMs) that are running on it.

Vulnerabilities in Drivers and Firmware: Type 1 hypervisors need specialized drivers and firmware parts to access hardware. These components' vulnerabilities may be used to access the underlying system.

Escape from a Guest VM: Although Type 1 hypervisors aim to provide high isolation between guest VMs, flaws in the hypervisor or hardware could theoretically allow an attacker to access the hypervisor or other VMs via escaping from a guest VM.

Strategies for Mitigation:

Regular Patching and Updates: To reduce known vulnerabilities, keep the hypervisor, drivers, and firmware up to date with the most recent security patches.

Use hardware characteristics that give additional layers of security and isolation across virtual machines, such as Intel VT-x and AMD-V.

Regularly check the hypervisor and its components for vulnerabilities via security audits and penetration testing. Potential flaws can be found with the aid of penetration testing.

Reduce the attack surface by disabling unused features and components in a minimal hypervisor configuration.

Network Segmentation: To lessen the impact of potential breaches, use virtual LANs (VLANs) or network segmentation.

Type 2 hypervisors (User-Space): In comparison to Type 1 hypervisors, Type 2 hypervisors operate inside a traditional operating system and have a greater attack surface. They are typically utilized in testing, development, and non-production situations.

Vulnerabilities:

Vulnerabilities in the host operating system can negatively affect the security of the hypervisor and all guest virtual machines.

Inter-VM Attacks: Because Type 2 hypervisors depend on the host OS for a number of their operations, an attacker may be able to target guest VMs by exploiting flaws in the host OS.

Resource Conflict: Resource sharing between the host OS and the guest VMs may cause conflict, which could have an impact on performance and security.

Privilege Escalation: If the host OS is compromised, an attacker may be able to take control of the hypervisor and the guest VMs through privilege escalation attacks.

Strategies for Mitigation:

Hardening the host operating system (OS) involves implementing security best practices such as routine patching, turning off superfluous services, and utilizing security technologies like firewalls and intrusion detection systems.

Isolation Mechanisms: To keep guest VMs apart from one another and from the host OS, use the security tools offered by the hypervisor.

Regular Backups: To recover in the event of a compromise, keep current backups of critical guest VMs.

fewer use cases Due to their intrinsic reliance on the host OS, type 2 hypervisors are often less suitable for high-security situations. Only use them in controlled, non-critical contexts, if possible.

Network segmentation should be used to reduce the impact of potential attacks on guest VMs.

1. **Confirm that you have stopped and terminated your AWS Linux server instance. To confirm, simply type your name below.**

Dennis Sarpong

1. *Link: https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html* [↑](#footnote-ref-1)