Workshop 4 Cloud computing

Introduction

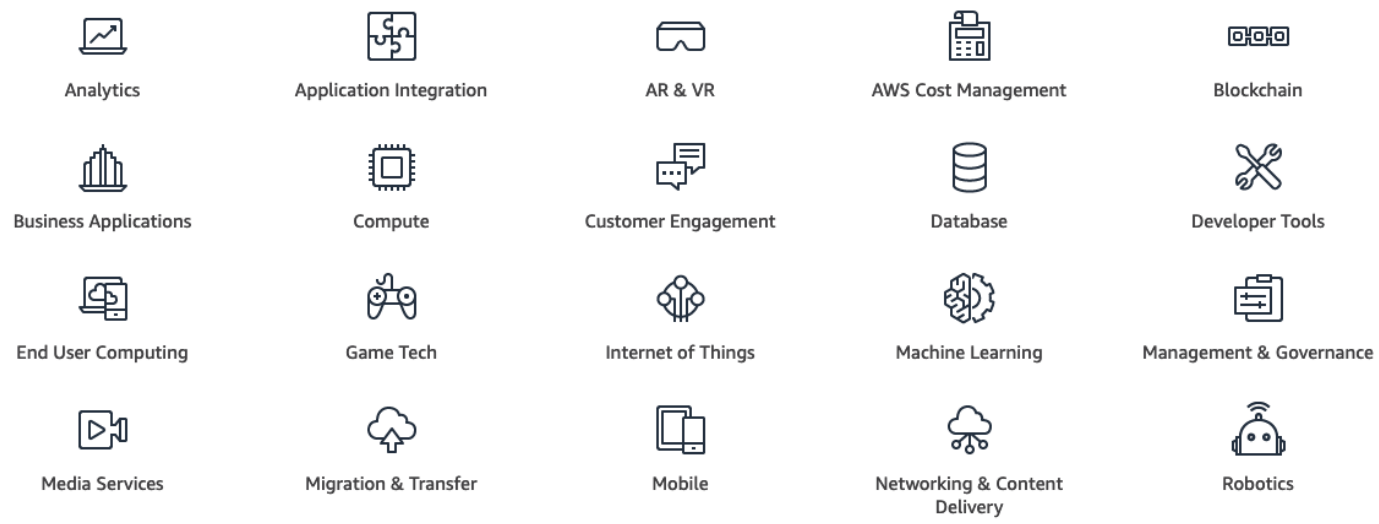
Cloud computing is the on-demand availability of computer system resources, especially data storage and computing power. The term is generally used to describe data centers available to many users over the Internet. Large clouds often have functions distributed over multiple locations from central servers.

Service providers... 100s of service providers, but releability and services vary greatly.

- Vultr
- Kamatera
- Digital Ocean
- ...
- IBM Cloud
- Microsoft Azure
- Google Cloud
- Amazon Web Services

Amazon Web Services (as an example)

As most of the Cloud computing provider their offer a wide varity of services: From generic "computer time" and "storage" to specilist VR or IA (machine learning)



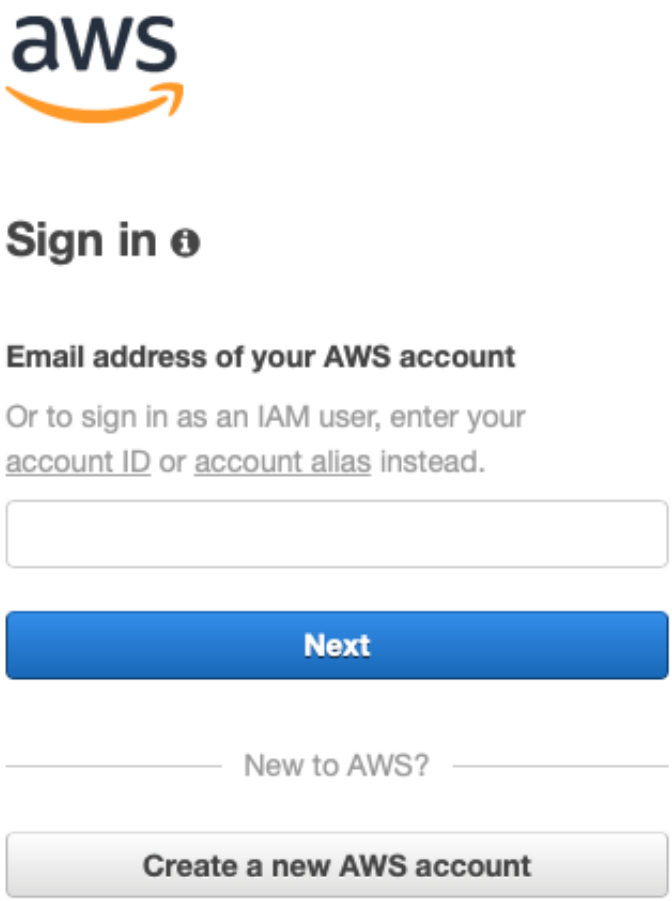
Bioinformatics is a fast evolving fields with bispocked processes, some can be automoted (e.g. galaxy platform), many are one of and need clean enviromenent and computer time.

We need a remote/virtual machine that will run nicely. Nothing less, nothing more.

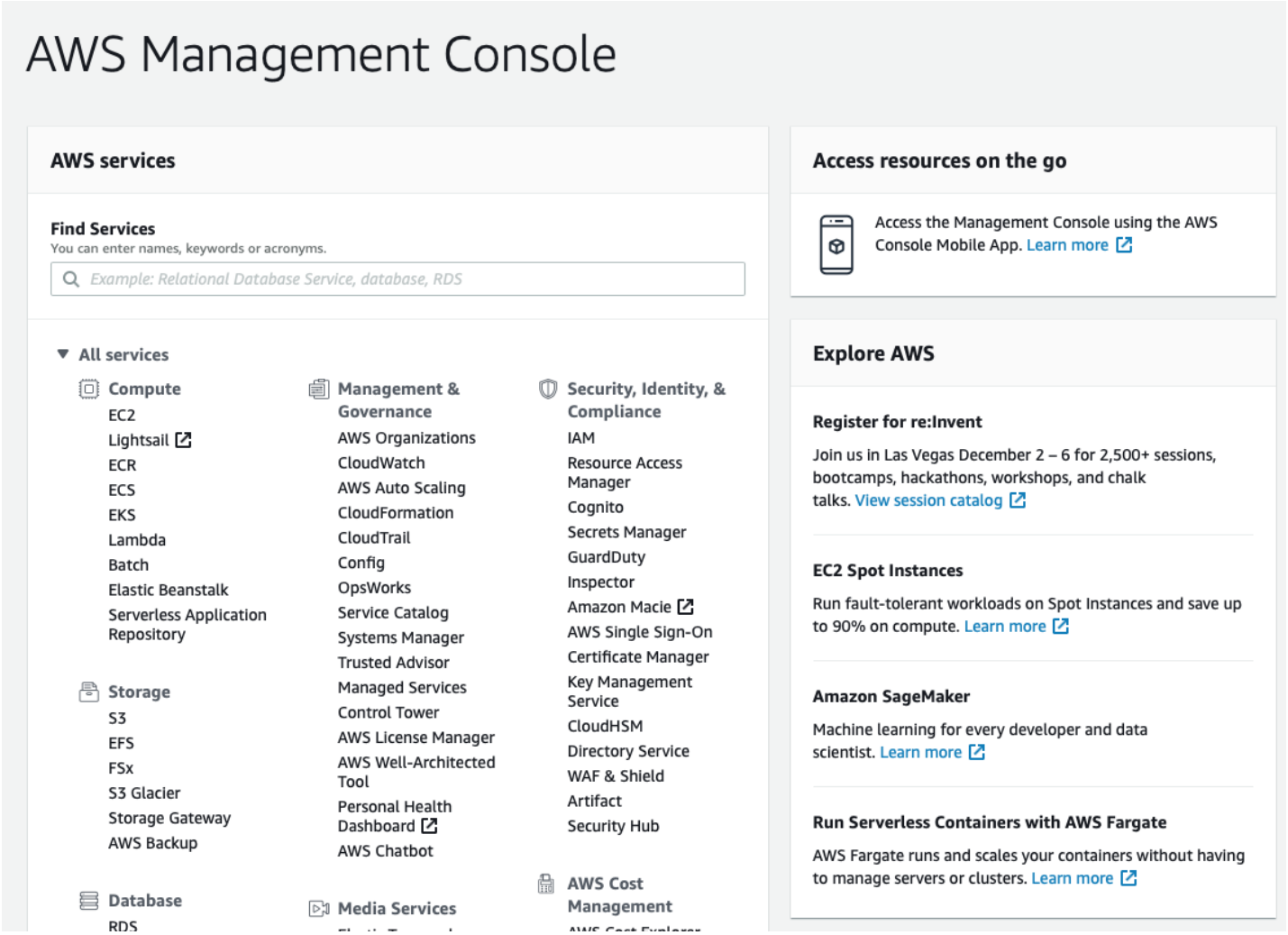
1. Login to your AWS account
2. Creating and Launching your EC2 Instance
3. Access your new cloud computer (remote machine)

Login to your AWS account

- Login (or create an account [with a credit card])



- Access EC2 (Elastic Compute Cloud)



Creating and Launching your EC2 Instance

Step 1: Launch an Instance

You can launch a Linux instance using the AWS Management Console as described in the following procedure. This tutorial is intended to help you launch your first instance quickly, so it doesn't cover all possible options. For more information about the advanced options, see Launching an Instance.

1. Open the Amazon EC2 console at https://console.aws.amazon.com/ec2/.
2. From the console dashboard, choose **Launch Instance**.
3. The **Choose an Amazon Machine Image (AMI)** page displays a list of basic configurations, called *Amazon Machine Images (AMIs)*, that serve as templates for your instance. Select an "Ubuntu 18.04". Notice that these AMIs are marked "Free tier eligible."
4. On the **Choose an Instance Type** page, you can select the hardware configuration of your instance. Select the `t2.micro` type, which is selected by default. Notice that this instance type is eligible for the free tier.

5. Choose **Review and Launch** to let the wizard complete the other configuration settings for you.

6. On the **Review Instance Launch** page, choose **Launch**.

7. When prompted for a key pair, select **Create a new key pair**, enter a name for the key pair, and then choose **Download Key Pair**. This is the only chance for you to save the private key file, so be sure to download it. Save the private key file in a safe place. You'll need to provide the name of your key pair when you launch an instance and the corresponding private key each time you connect to the instance.

   When you are ready, select the acknowledgement check box, and then choose **Launch Instances**.

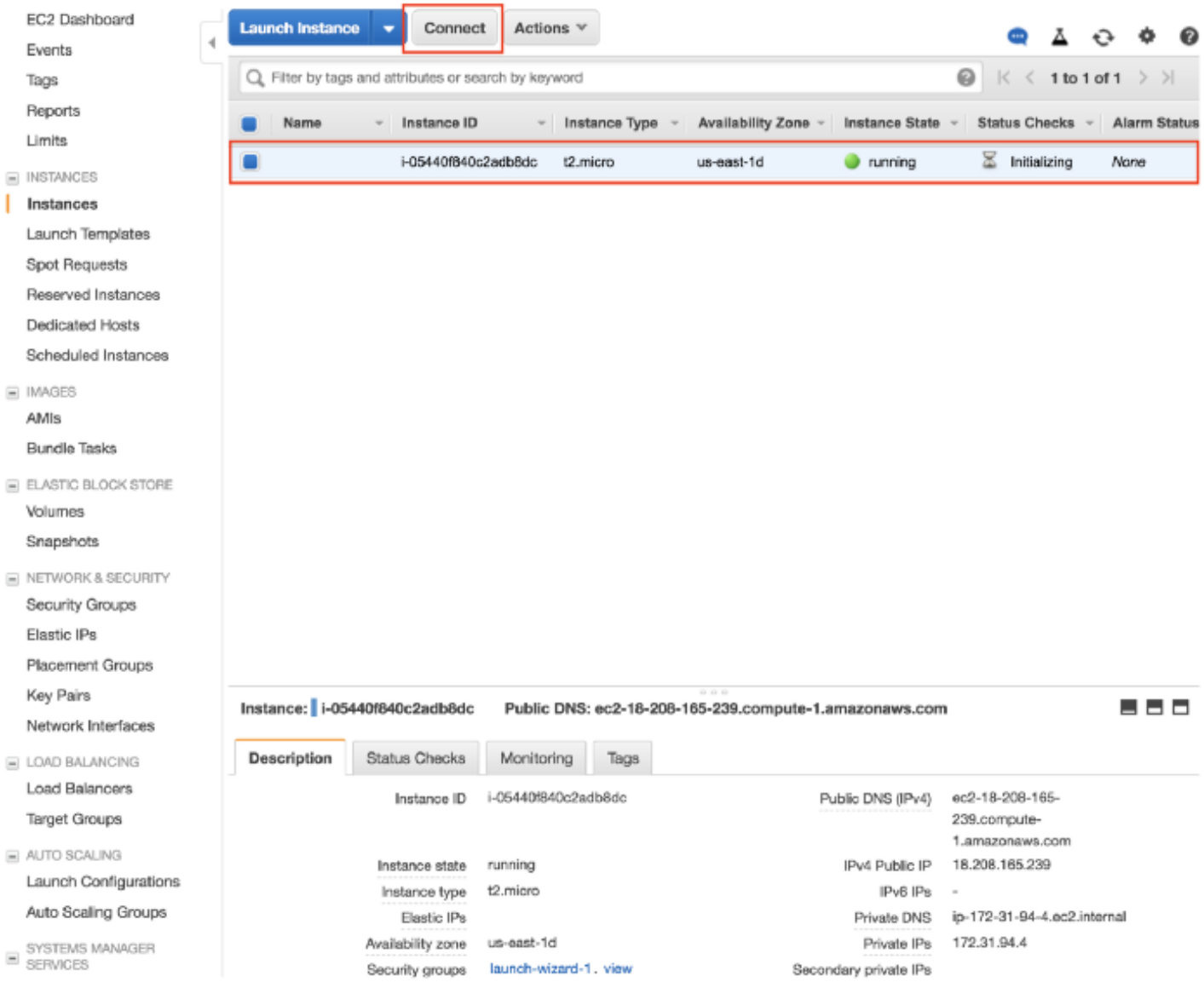
8. A confirmation page lets you know that your instance is launching. Choose **View Instances** to close the confirmation page and return to the console.

9. On the **Instances** screen, you can view the status of the launch. It takes a short time for an instance to launch. When you launch an instance, its initial state is `pending`. After the instance starts, its state changes to `running` and it receives a public DNS name. (If the **Public DNS (IPv4)** column is hidden, choose **Show/Hide Columns** (the gear-shaped icon) in the top right corner of the page and then select **Public DNS (IPv4)**.)

10. It can take a few minutes for the instance to be ready so that you can connect to it. Check that your instance has passed its status checks; you can view this information in the **Status Checks** column.

Step 2: Connect to Your Instance

You can't connect to your instance unless you launched it with a key pair for which you have the `.pem` file.



1. In a terminal window, use the ssh command to connect to the instance. You specify the private key (`.pem`) file, the user name for your AMI, and the public DNS name for your instance. For example, if you used Ubuntu, the user name is `ubuntu` .

   ```
   ssh -i /path/my-key-pair.pem ubuntu@ec2-198-51-100-1.compute-1.amazonaws.com
   ```

2. You see a response like the following:

   ```
   The authenticity of host 'ec2-198-51-100-1.compute-1.amazonaws.com (10.254.142.3
   can't be established.
   RSA key fingerprint is 1f:51:ae:28:bf:89:e9:d8:1f:25:5d:37:2d:7d:b8:ca:9f:f5:f1:
   Are you sure you want to continue connecting (yes/no)?
   ```

Step 3: Clean Up Your Instance

After you've finished with the instance that you created for this class, you should clean up by terminating

the instance.

Important
Terminating an instance effectively deletes it; you can't reconnect to an instance after you've terminated it.

If you launched an instance that is not within the AWS Free Tier, you'll stop incurring charges for that instance as soon as the instance status changes to `shutting down` or `terminated`.

To terminate your instance

1. In the navigation pane, choose **Instances**. In the list of instances, select the instance.
2. Choose **Actions**, **Instance State**, **Terminate**.
3. Choose **Yes, Terminate** when prompted for confirmation.

   Amazon EC2 shuts down and terminates your instance. After your instance is terminated, it remains visible on the console for a short while, and then the entry is deleted.

Adapted from: Amazon EC2 Get Started