

## Cloud Computing task 1 Report Max Nummila

This is a report about task 1 in the cloud computing course. The report has screenshots to either show the work done or to support what is written in the report.



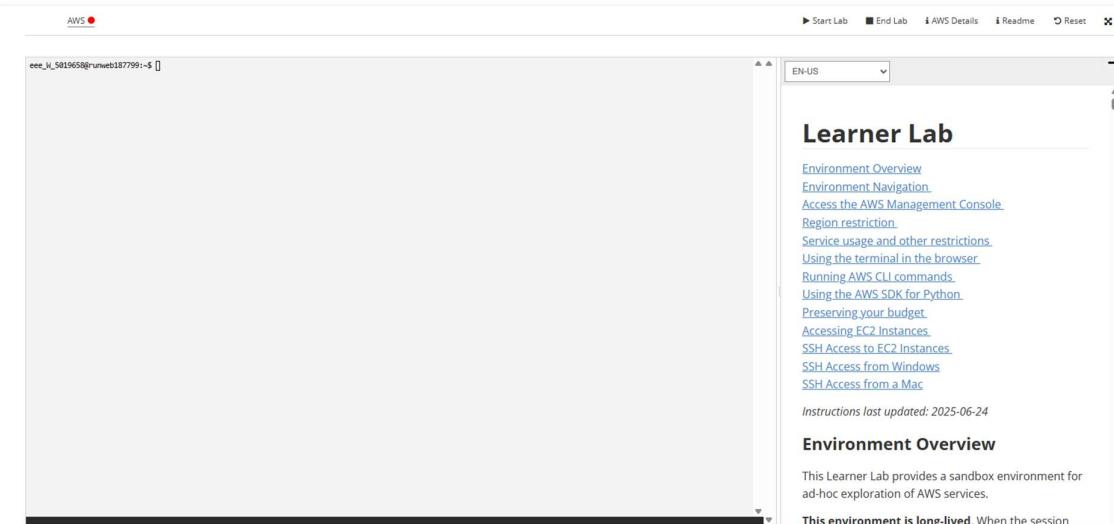
The screenshot shows the AWS Academy Learner Lab landing page. At the top, there's a navigation bar with links for Home, Modules, Discussions, Grades, and Learner Lab (Whiteboard). The main title is "AWS Academy Learner Lab [130480]". Below the title is a decorative graphic featuring a stylized building, clouds, and the AWS logo. A detailed description of the service follows, mentioning it's a long-running sandbox environment for ad-hoc exploration of AWS services. It notes that students will have access to a restricted set of AWS services, and some documentation may not work. It also cautions against running certain services like EC2 instances to prevent charges. Session details are provided: each session lasts 4 hours by default, and users can extend it by pressing the start button. At the end of each session, resources are automatically shut down. The page ends with a "Get Started" button and a note that it's for educator-designed project work.

### Step 1: Login on the AWS console

Went to modules since the front page told me to get started there.



This popped up as a probable next step since it launches the lab.



The screenshot shows the AWS Academy Learner Lab environment. At the top, there's a toolbar with buttons for Start Lab, End Lab, AWS Details, Readme, and Reset. The main area is titled "Learner Lab" and contains a sidebar with links to "Environment Overview", "Environment Navigation", "Access the AWS Management Console", "Region restriction", "Service usage and other restrictions", "Using the terminal in the browser", "Running AWS CLI commands", "Using the AWS SDK for Python", "Preserving your budget", "Accessing EC2 Instances", "SSH Access to EC2 Instances", "SSH Access from Windows", and "SSH Access from a Mac". Below the sidebar, there's a message about instructions being updated. Further down, there's a section titled "Environment Overview" with a note that the environment is long-lived. The bottom of the screen shows a terminal window with the command "eee\_u\_5019653@runweb187799:~\$".

After agreeing to some terms I got to where I need to be, then I started it with the button

labeled as such. The lab started after a short wait and next was finding the console. The readme told me to press the AWS with the green button and so I did.

The screenshot shows the AWS Console Home page. It includes sections for Recently visited services (with a placeholder for a recently visited service), Applications (empty), Welcome to AWS (with links for Getting started with AWS and Training and certification), AWS Health (showing 0 open issues and 0 scheduled changes), and Cost and usage (showing current month cost of \$0.00 and a forecasted bar chart for the next 12 months).

Okay, it says console home so step 1 done.

## Step 2: Launch an instance via the console

First thought is to press the EC2 since we talked about it on the lecture.

The screenshot shows the AWS EC2 Dashboard. The left sidebar lists navigation options: Dashboard, Instances, Images, Elastic Block Store, Network & Security, and Help. The main content area displays EC2 resources (0 instances running, 0 auto scaling groups, etc.) and a "Launch instance" section with "Launch instance" and "Migrate a server" buttons. It also shows instance alarms and scheduled events. On the right, there's a "Service health" section indicating the service is operating normally, and a "Zones" table listing availability zones (us-east-1a through us-east-1e) and their zone IDs.

And there it is “Launch instance”.

I named it TestLab1 and picked Amazon linux as it is the default, picked t2.micro as it was recommended in the instructions. I created a keypair and made a backup in case of emergency, since incase I would not have the key, I wouldn't be able to SSH to the VM. I left all the network and storage settings except for the HTTP one, which I allowed.

The screenshot shows a green success banner at the top stating "Success Successfully initiated launch of instance i-04876978a5e64da40". Below this is a "Launch log" button. The main area is titled "Next Steps" with a search bar. It contains ten cards arranged in two rows of five:

- Create billing usage alerts**: To manage costs and avoid surprise bills, set up email notifications for billing usage thresholds. Includes a "Create billing alerts" button.
- Connect to your instance**: Once your instance is running, log into it from your local computer. Includes a "Connect to instance" button.
- Connect an RDS database**: Configure the connection between an EC2 instance and a database to allow traffic flow between them. Includes a "Connect an RDS database" button.
- Create EBS snapshot policy**: Create a policy that automates the creation, retention, and deletion of EBS snapshots. Includes a "Create EBS snapshot policy" button.
- Manage detailed monitoring**: Enable or disable detailed monitoring for the instance. If you enable detailed monitoring, the Amazon EC2 console displays monitoring graphs with a 1-minute period. Includes a "Manage detailed monitoring" button.

- Create AWS budget**: AWS Budgets allows you to create budgets, forecast spend, and take action on your costs and usage from a single location. Includes a "Create AWS budget" button.
- Manage CloudWatch alarms**: Create or update Amazon CloudWatch alarms for the instance. Includes a "Manage CloudWatch alarms" button.
- Disaster recovery for your instances**: Recover the instances you just launched into a different Availability Zone or a different Region using AWS Elastic Disaster Recovery (DRS). Includes a "Discover resources for your instances" button.
- Monitor for suspicious runtime activities**: Amazon GuardDuty enables you to continuously monitor for malicious runtime activity and unauthorized behavior, with near real-time visibility into on-host activities occurring across your Amazon EC2 workloads.
- Get instance screenshot**: Capture a screenshot from the instance and view it as an image. This is useful for troubleshooting an unreachable instance. Includes a "Get instance screenshot" button.

Success, step 2 done.

Step 3: Check the public IP address and public DNS of your newly created VM. Then, using SSH and your key file, connect to your remote VM from your local machine.

So, to find my instance I went back to my instances, and I can already see the public DNS and ipv4. The server is probably somewhere in North Virginia, as that is my region on AWS but I am not sure.

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 ...
TestLab1	i-04876978a5e64da40	Running	t2.micro	Initializing	View alarms +	us-east-1b	ec2-54-161-255-172.co...	54.161.255.172

Then I use gitbash to find my pem file for the key, which I then chmod 400 to make sure it works in gitbash and is not too open. Then I ssh to it with my key and the ip.

```
maxnu@VitPC MINGW64 ~
$ ssh -i ~/Downloads/max-awsacademy.pem ec2-user@54.161.255.172
The authenticity of host '54.161.255.172 (54.161.255.172)' can't be established.
ED25519 key fingerprint is SHA256:rC9K/vC5vEhC/mk8fC5F2ygCErNqYaiPbZAyOoHYGzc.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '54.161.255.172' (ED25519) to the list of known hosts

      ,      #
      ~\_\_ #####      Amazon Linux 2023
      ~~ \####\_
      ~~ \###|
      ~~ \#/ __ https://aws.amazon.com/linux/amazon-linux-2023
      ~~ V~' '-->
      ~~ /_
      ~~ ._. /_
      ~~ /_/
      ~~ /m/'

[ec2-user@ip-172-31-83-218 ~]$ |
```

It took a couple of minutes but now it is live.

#### Step 4: Check if your Linux distribution contains packages to be updated

```
[ec2-user@ip-172-31-83-218 ~]$ sudo yum update
Amazon Linux 2023 Kernel Livepatch repository                               200 kB/s | 21 kB     00:00
=====
WARNING:
  A newer release of "Amazon Linux" is available.

  Available Versions:

  Version 2023.8.20250908:
    Run the following command to upgrade to 2023.8.20250908:
      dnf upgrade --releasever=2023.8.20250908

  Release notes:
    https://docs.aws.amazon.com/linux/al2023/release-notes/relnotes-2023.8.20250908.html

=====
Dependencies resolved.
Nothing to do.
Complete!
[ec2-user@ip-172-31-83-218 ~]$
```

Nothing to do, great!

#### Step 5: Check and report the following information

CPU Model name: Intel Xeon CPU E5-2686 v4

Cache size: L1d 32 KiB, L1i 32 KiB, L2 256 KiB, L3 45 MiB

CPU Clock Speed: 2.30 GHz

CPU Vendor: GenuineIntel

Name of Hypervisor vendor: Xen

```
CPU op-mode(s):          32-bit, 64-bit
Address sizes:           46 bits physical, 48 bits virtual
Byte Order:              Little Endian
CPU(s):
  On-line CPU(s) list:  0
Vendor ID:               GenuineIntel
Model name:              Intel(R) Xeon(R) CPU E5-2686 v4 @ 2.30GHz
CPU family:              6
Model:                  79
Thread(s) per core:     1
Core(s) per socket:     1
Socket(s):              1
Stepping:                1
BogoMIPS:                4600.00
Flags:
Virtualization features:
  Hypervisor vendor:   Xen
  Virtualization type: full
Caches (sum of all):
  L1d:                 32 KiB (1 instance)
  L1i:                 32 KiB (1 instance)
  L2:                  256 KiB (1 instance)
  L3:                  45 MiB (1 instance)
NUMA:
  NUMA node(s):         1
  NUMA node0 CPU(s):    0
```

Step 6: Still from your VM, execute the following command line

After missing a space I completed this

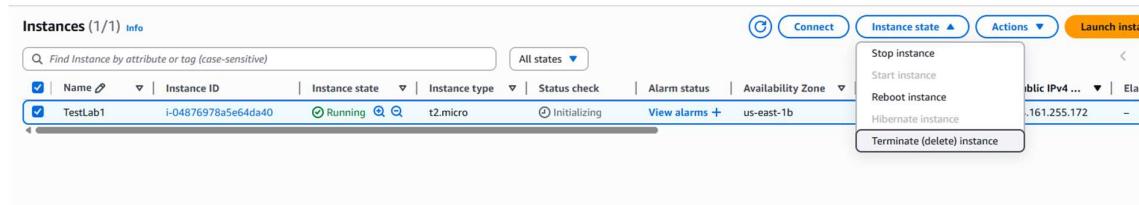
```
[ec2-user@ip-172-31-83-218 ~]$ curl"vm4460.kaj.pouta.csc.fi/logs.php?name=Max_Nummila">log.dat
-bash: curlvm4460.kaj.pouta.csc.fi/logs.php?name=Max_Nummila: No such file or directory
[ec2-user@ip-172-31-83-218 ~]$ curl "vm4460.kaj.pouta.csc.fi/logs.php?name=Max_Nummila">log.dat
% Total    % Received % Xferd  Average Speed   Time   Time   Current
          Dload Upload Total Spent   Left Speed
100  127  100  127  0    0   314     0  --::--  --::--  --::-- 314
[ec2-user@ip-172-31-83-218 ~]$
```

Step 7: download the created log.dat file on your computer

```
maxnu@VitPC MINGW64 /
$ scp -i ~/Downloads/max-awsacademy.pem ec2-user@54.161.255.172:~/log.dat ~/Downloads/
log.dat                                         100%  127      0.5KB/s  00:00
```

Easy enough with the scp command

Step 8: You are done with the VM! Remember to kill your VM!



That should do it.

#### Step 9: log.dat file

Name: Max\_Nummila -- 54.161.255.172 -- ec2-54-161-255-172.compute-1.amazonaws.com -- 192.168.1.14 -- 1757518539 -- curl/8.11.1

#### Reflection

Have you learned anything completely new?

Yes, I have studied cloud computing a bit during my own time but I hadn't actually used any AWS services before.

Did anything surprise you?

The layout of the Amazon services were more user and beginner friendly than I expected, which was a pleasant surprise.

Did you find anything challenging? Why?

-

Did you find anything satisfying? Why?

Completing the tasks, felt nice actually doing hands on learning again after so much theory last year, and during my own free time learning about the topic.