# Laboratory Practice Report

## Practice 6

October 11, 2023

Computer Systems Engineering

*Cloud Architecture*

Prof. M.S. Rodolfo Luthe Ríos

Marco Ricardo Cordero Hernández

is727272@iteso.mx

ITESO
Universidad Jesuita
de Guadalajara

## Abstract

The current work demonstrates the usage of multiple availability ensuring technologies over a cloud provider, in which applications and services are ensured to be accessible and usable through prolonged time lapses.

The contents of the following demonstration might not seem extensive, however, all that's been learned to this moment in cloud knowledge, discovery and practice it's being implemented in some way or another.

Other skills such as diagram creation for implementation planning it's practiced as well. This could also be somewhat irrelevant, but, overlooking and underestimating this step in a real world scenario would potentially determine the project's success.

## State of the Art

Commodities have always been something desirable in people's life. When something goes wrong, a quick response has to be deployed in order to restore common order, or at least going back to what was normal before. This topic it's applicable to several areas such as healthcare, business, production lines, lifestyles, etc. As it can be inferred, information technology isn't exempt from this search of desirable normal operation means, and such, several techniques of achieving seamless order have been developed to address this requirement.

Although the concept of high availability of resources and mitigation plans after prevention have always been something present in all kinds of projects, cloud automatic scaling for computing instances and such is a relatively new technology. In fact, Amazon Web Services was the first major cloud vendor to introduce scalability for automatic instance creation [1], making AWS Auto Scaling service [2] an integral part of EC2 [3] offerings.

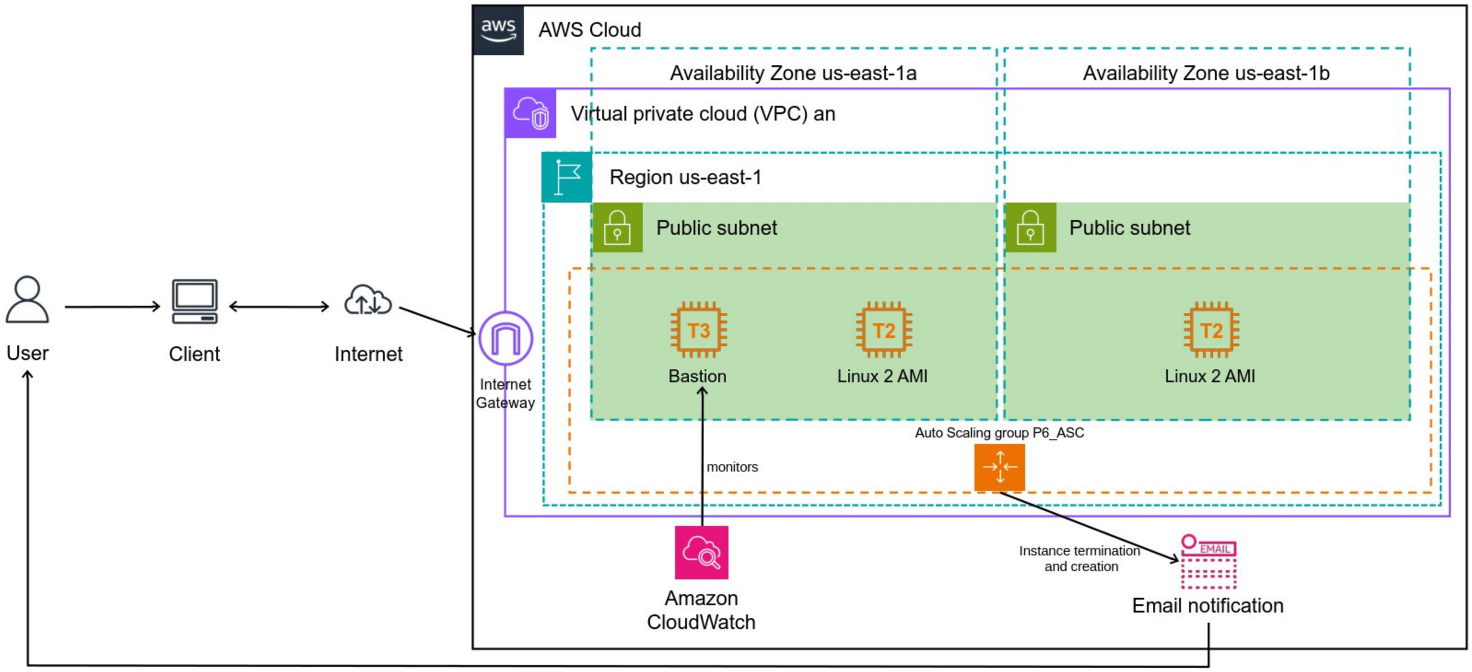
The core idea behind any auto scaling technology (not only AWS') depicts the necessity of having functional times at its optimal condition, meaning that downtimes *need* to be from zero to none. This is real for multiple applications like banking portals, hospital information systems, trading platforms, and many other cases where stall executions result in millionaire losses.

Even though vertical scaling might seem lost and overtook by horizontal one [4], the fact is that both environment resource allocation/growth can take place within the same architecture. Take for example the scenario in which an instance has a determined computing capacity, horizontal scaling can be seen at the moment of CPU overpassing, where the initial instance can't handle more transactions in some time frame and potential application lose is at a stake. Conveniently, a predefined auto scaling rule would've been set to deploy a helper secondary instance, and the incoming traffic would temporarily be redirected to this new element. In the meantime, vertical scaling would take place inside the same first instance, in which CPU specs could be enhanced or some other action.

In this practice, the aforementioned scaling technology belonging to AWS will be used to demonstrate its capabilities and how it can be used in further application and services architectures.

## Diagram

The following architecture it's proposed as a graphic solution for the stated goals.

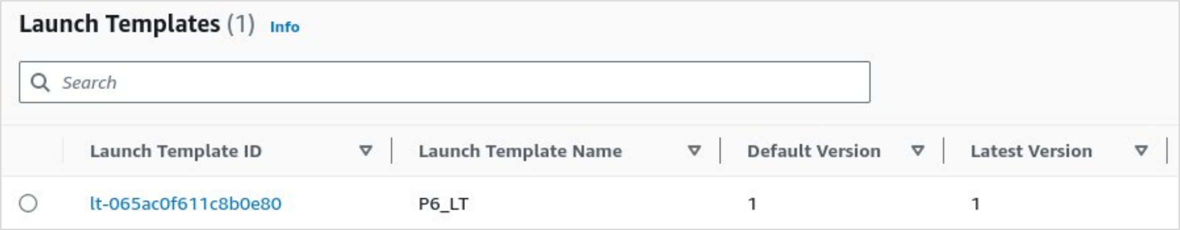


## Practice Development

For the development of this practice, the AWS Auto Scaling service will be used to demonstrate the usage and benefits of automatically deployed nodes on an architecture.

First, and EC2 launch template has to be created with the following requirements:

- Check EC2 Auto Scaling guidance option
- OS: Amazon Linux 2 AMI
- Instance type: t2.micro
- Login: provided key pair
- Network:
  - New security group
- Storage: 30gb



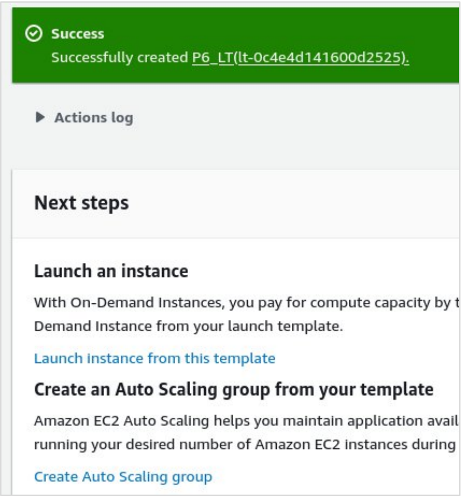
**Launch Templates** (1) Info

Search

| | Launch Template ID | Launch Template Name | Default Version | Latest Version |
|---|---|---|---|---|
| ○ | lt-065ac0f611c8b0e80 | P6_LT | 1 | 1 |

The purpose of this template is to automatically create a new instance to aid in the auto scaling process.

Now, the auto scaling group itself has to be created as well. A shortcut to create will be prompted after the template (successful) creation.



**Success**
Successfully created P6_LT(lt-0c4e4d141600d2525).

▶ Actions log

**Next steps**

**Launch an instance**

With On-Demand Instances, you pay for compute capacity by t
Demand Instance from your launch template.

Launch instance from this template

**Create an Auto Scaling group from your template**

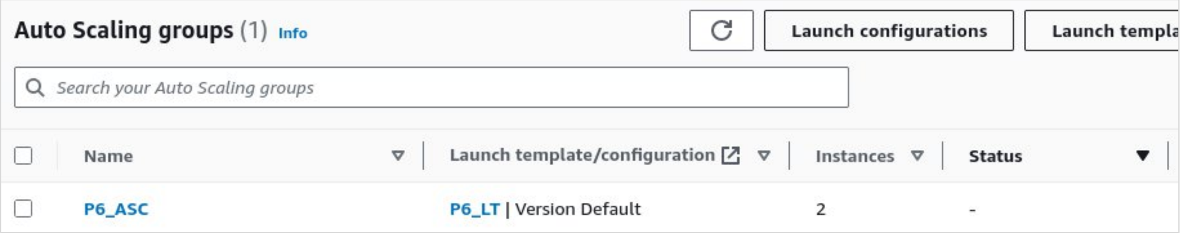
Amazon EC2 Auto Scaling helps you maintain application avail
running your desired number of Amazon EC2 instances during

Create Auto Scaling group

By accessing the creation through this method, some fields will be already filled with the template information, however, these requirements are also needed:

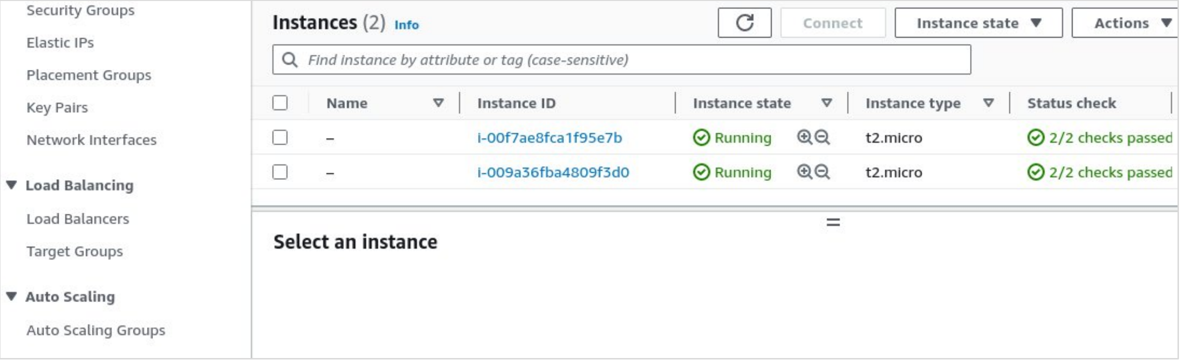
- AN VPC
- Two different public subnets in different availability zones

- No load balancing
- Default health checks
- Scaling policies:
  - Capacity: 2 (in different availability zones)
  - Between 2 and 4 instances
  - Deploy: CPU usage > 50%
- Notify of launch and terminate events on any desired email
  - is727272@iteso.mx for this practice



| □ | Name | Launch template/configuration | Instances | Status |
|---|---|---|---|---|
| □ | P6_ASC | P6_LT \| Version Default | 2 | - |

Even though template creation suggested this service, it can also be accessed through the EC2 section. In fact, EC2 instance creation has to be reviewed (there should be at least 2 instances) and auto scaling groups can be seen at the bottom.

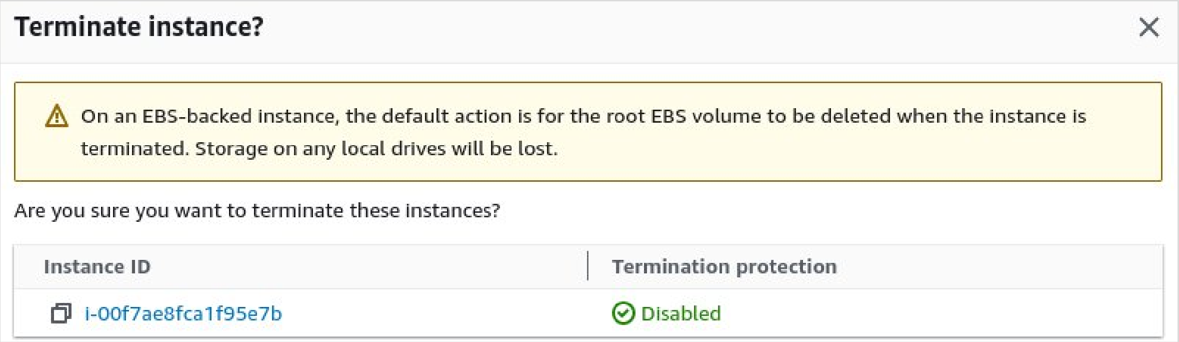


| □ | Name | Instance ID | Instance state | Instance type | Status check |
|---|---|---|---|---|---|
| □ | – | i-00f7ae8fca1f95e7b | ✓ Running | t2.micro | ✓ 2/2 checks passed |
| □ | – | i-009a36fba4809f3d0 | ✓ Running | t2.micro | ✓ 2/2 checks passed |

Select an instance

These nameless instances has been automatically created by the auto scaling group, this is because 2 minimum instances had been defined.

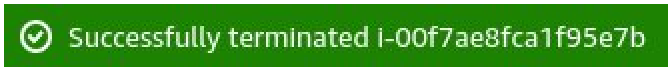
Once the previous steps are done, one instance (top one) will be *terminated* to demonstrate the auto scaling groups capabilities.


Terminate instance?

⚠ On an EBS-backed instance, the default action is for the root EBS volume to be deleted when the instance is terminated. Storage on any local drives will be lost.

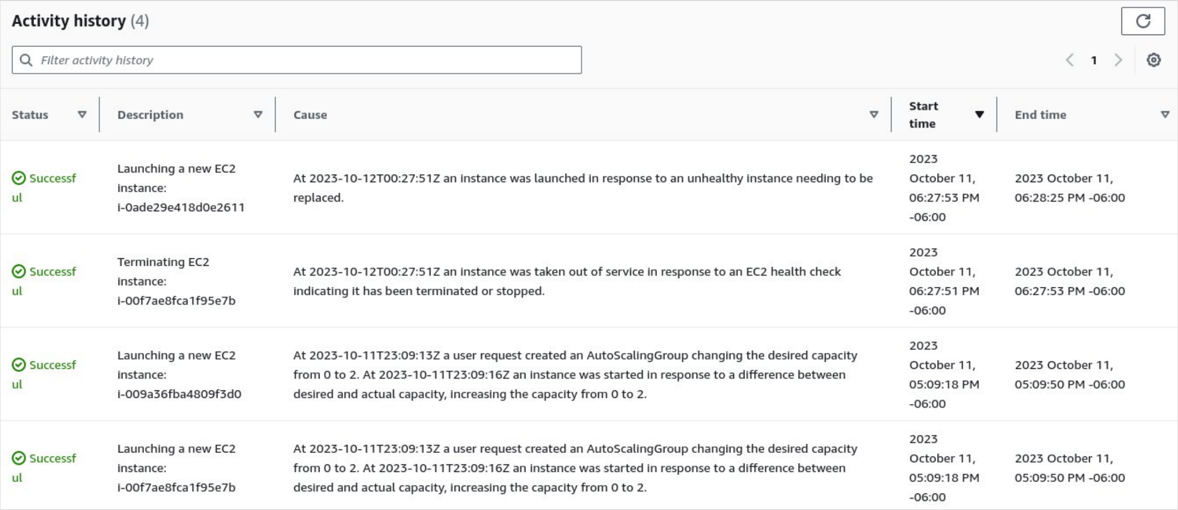
Are you sure you want to terminate these instances?

| Instance ID | Termination protection |
|---|---|
| i-00f7ae8fca1f95e7b | ✓ Disabled |


✓ Successfully terminated i-00f7ae8fca1f95e7b

To review activity from the auto scaling group, a dedicated section can be accessed through the same console apartment (activity and instance management).

First, activity history shows both termination and new instance launch events, along with the previous two events. Default warming time was set to 300 seconds, so it may take a while until this activity is shown.

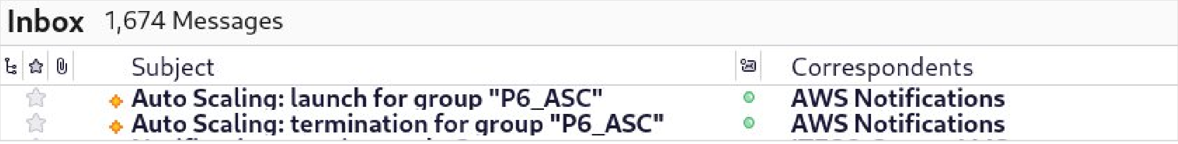


**Activity history** (4)

Filter activity history

| Status | Description | Cause | Start time | End time |
|---|---|---|---|---|
| ✓ Successful | Launching a new EC2 instance: i-0ade29e418d0e2611 | At 2023-10-12T00:27:51Z an instance was launched in response to an unhealthy instance needing to be replaced. | 2023 October 11, 06:27:53 PM -06:00 | 2023 October 11, 06:28:25 PM -06:00 |
| ✓ Successful | Terminating EC2 instance: i-00f7ae8fca1f95e7b | At 2023-10-12T00:27:51Z an instance was taken out of service in response to an EC2 health check indicating it has been terminated or stopped. | 2023 October 11, 06:27:51 PM -06:00 | 2023 October 11, 06:27:53 PM -06:00 |
| ✓ Successful | Launching a new EC2 instance: i-009a36fba4809f3d0 | At 2023-10-11T23:09:13Z a user request created an AutoScalingGroup changing the desired capacity from 0 to 2. At 2023-10-11T23:09:16Z an instance was started in response to a difference between desired and actual capacity, increasing the capacity from 0 to 2. | 2023 October 11, 05:09:18 PM -06:00 | 2023 October 11, 05:09:50 PM -06:00 |
| ✓ Successful | Launching a new EC2 instance: i-00f7ae8fca1f95e7b | At 2023-10-11T23:09:13Z a user request created an AutoScalingGroup changing the desired capacity from 0 to 2. At 2023-10-11T23:09:16Z an instance was started in response to a difference between desired and actual capacity, increasing the capacity from 0 to 2. | 2023 October 11, 05:09:18 PM -06:00 | 2023 October 11, 05:09:50 PM -06:00 |

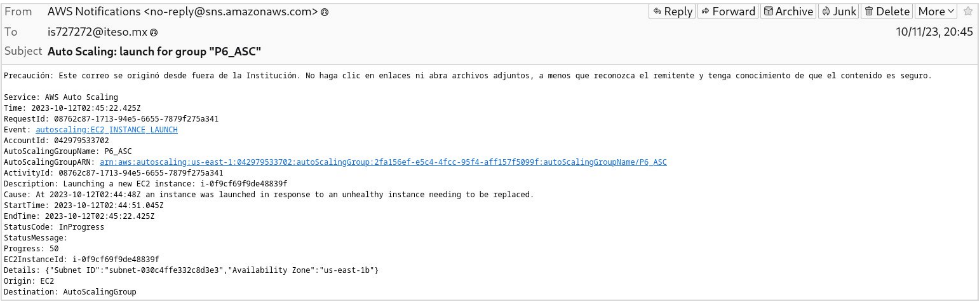
As such events have been configured to send notifications through email, this can also be verified in any email client.



**Inbox** 1,674 Messages

| Subject | Correspondents |
|---|---|
| Auto Scaling: launch for group "P6_ASC" | AWS Notifications |
| Auto Scaling: termination for group "P6_ASC" | AWS Notifications |

The contents of these mails show information of both events, correspondingly.



From AWS Notifications <no-reply@sns.amazonaws.com>

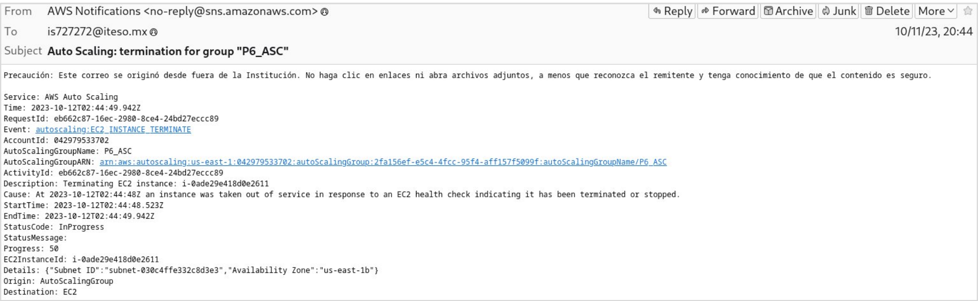
To is727272@iteso.mx

10/11/23, 20:45

Subject **Auto Scaling: launch for group "P6_ASC"**

```
Precaución: Este correo se originó desde fuera de la Institución. No haga clic en enlaces ni abra archivos adjuntos, a menos que reconozca el remitente y tenga conocimiento de que el contenido es seguro.

Service: AWS Auto Scaling
Time: 2023-10-12T02:45:22.425Z
RequestId: 08762c87-1713-94e5-6655-7879f275a341
Event: autoscaling:EC2_INSTANCE_LAUNCH
AccountId: 042979533702
AutoScalingGroupName: P6_ASC
AutoScalingGroupARN: arn:aws:autoscaling:us-east-1:042979533702:autoScalingGroup:2fa156ef-e5c4-4fcc-95f4-aff157f5099f:autoScalingGroupName/P6_ASC
ActivityId: 08762c87-1713-94e5-6655-7879f275a341
Description: Launching a new EC2 instance: i-0f9cf69f9de48839f
Cause: At 2023-10-12T02:44:48Z an instance was launched in response to an unhealthy instance needing to be replaced.
StartTime: 2023-10-12T02:44:51.045Z
EndTime: 2023-10-12T02:45:22.425Z
StatusCode: InProgress
StatusMessage:
Progress: 50
EC2InstanceId: i-0f9cf69f9de48839f
Details: {"Subnet ID":"subnet-030c4ffe332c8d3e3","Availability Zone":"us-east-1b"}
Origin: EC2
Destination: AutoScalingGroup
```



From AWS Notifications <no-reply@sns.amazonaws.com>

To is727272@iteso.mx

10/11/23, 20:44

Subject **Auto Scaling: termination for group "P6_ASC"**

```
Precaución: Este correo se originó desde fuera de la Institución. No haga clic en enlaces ni abra archivos adjuntos, a menos que reconozca el remitente y tenga conocimiento de que el contenido es seguro.

Service: AWS Auto Scaling
Time: 2023-10-12T02:44:49.942Z
RequestId: eb662c87-16ec-2980-8ce4-24bd27eccc89
Event: autoscaling:EC2_INSTANCE_TERMINATE
AccountId: 042979533702
AutoScalingGroupName: P6_ASC
AutoScalingGroupARN: arn:aws:autoscaling:us-east-1:042979533702:autoScalingGroup:2fa156ef-e5c4-4fcc-95f4-aff157f5099f:autoScalingGroupName/P6_ASC
ActivityId: eb662c87-16ec-2980-8ce4-24bd27eccc89
Description: Terminating EC2 instance: i-0ade29e418d0e2611
Cause: At 2023-10-12T02:44:48Z an instance was taken out of service in response to an EC2 health check indicating it has been terminated or stopped.
StartTime: 2023-10-12T02:44:48.523Z
EndTime: 2023-10-12T02:44:49.942Z
StatusCode: InProgress
StatusMessage:
Progress: 50
EC2InstanceId: i-0ade29e418d0e2611
Details: {"Subnet ID":"subnet-030c4ffe332c8d3e3","Availability Zone":"us-east-1b"}
Origin: AutoScalingGroup
Destination: EC2
```

After this demonstration, *auto scaling group has to be terminated*, otherwise, the service will continue to deploy new instances, and consequently, charge for them. For the automatically created instances, the termination action would also terminate them instantly (no manual intervention),

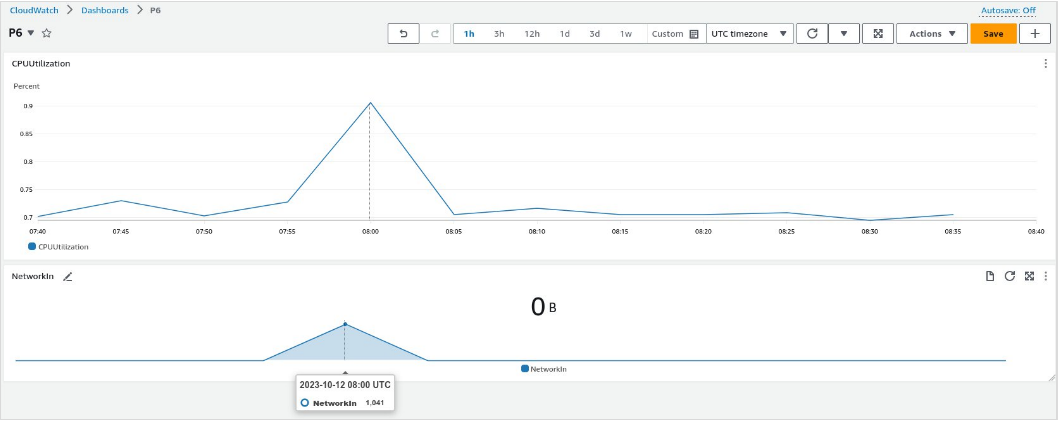
## Instance monitoring with CloudWatch

Is often conveniently the visualization of metrics and statistics through alternative means other than visual scrapping or such. For this, AWS provides the CloudWatch service, dedicated to monitor other services inside the same cloud through a dashboard style view.

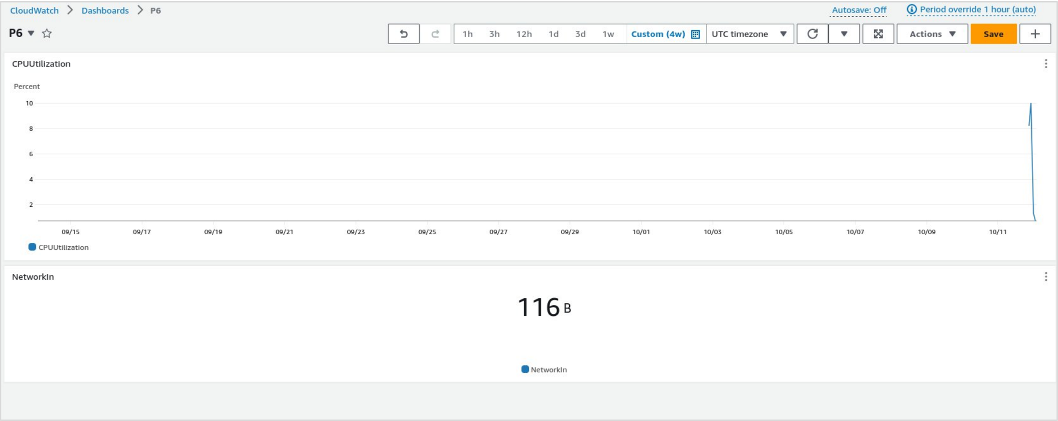
For this demonstration, the Windows admin (bastion) instance from the first practice will be used (Windows Server 2022, t3.small, 30gb). A pair of monitoring widgets will be utilized, with a visualization period of 1 hour and 4 weeks.

The final result would look like this:

1 hour view

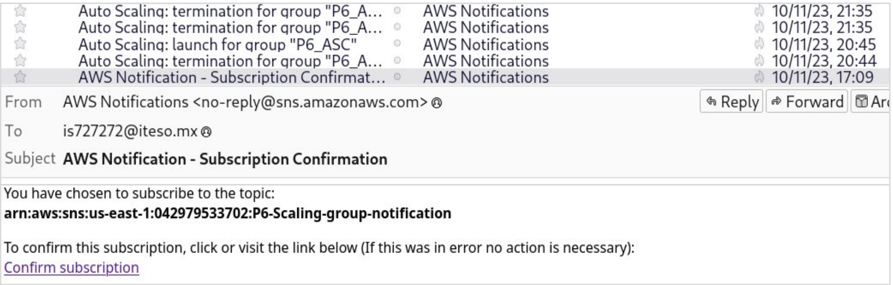


4 weeks view



## Problems and Solutions

For this practice, a single issue was found, this being not receiving instances events inside the auto scaling group. Even though it's intuitive, the mailing list subscription has to be accepted before getting any kind of alerts for events. Once the invitation has been agreed, new notifications should be received onwards (previous one are lost).



| | | |
|---|---|---|
| Auto Scaling: termination for group "P6_A… | AWS Notifications | 10/11/23, 21:35 |
| Auto Scaling: termination for group "P6_A… | AWS Notifications | 10/11/23, 21:35 |
| Auto Scaling: launch for group "P6_ASC" | AWS Notifications | 10/11/23, 20:45 |
| Auto Scaling: termination for group "P6_A… | AWS Notifications | 10/11/23, 20:44 |
| AWS Notification - Subscription Confirmat… | AWS Notifications | 10/11/23, 17:09 |

From AWS Notifications <no-reply@sns.amazonaws.com>

To is727272@iteso.mx

Subject **AWS Notification - Subscription Confirmation**

You have chosen to subscribe to the topic:
**arn:aws:sns:us-east-1:042979533702:P6-Scaling-group-notification**

To confirm this subscription, click or visit the link below (If this was in error no action is necessary):
Confirm subscription

## Experiments and Results

As this was a straightforward development, no experiments were conducted.

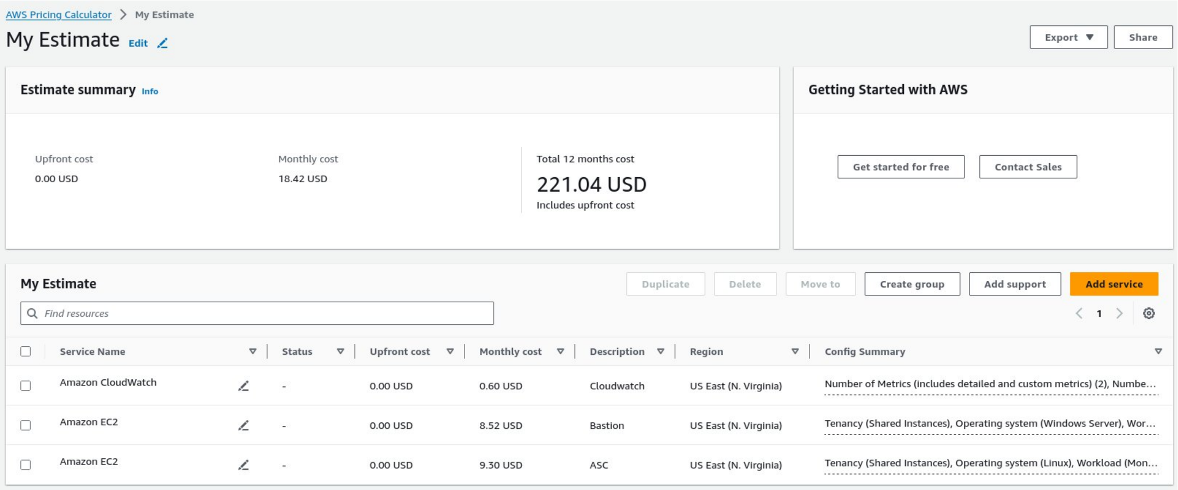
However, among the configurations of the created templates, one section stood out from others: load balancing. Although in this case this option wasn't selected, one real case scenario for its usage would be high concurrent applications that can fail when too many users utilize them at the same time. Automatic scaling service along with load balancers can mitigate this cases when configured correctly, combining techniques such as IP switching to new deployed instances, resulting in minimal downtime in common operations.

## Budget Justification

Taking into consideration

- One Windows server bastion instance on regular work weeks
- Two Linux instances with monthly spike traffic inside the auto scaling group
- CloudWatch utilization through a single dashboard

... The costs would be the following:



My Estimate

**Estimate summary**

Upfront cost
0.00 USD

Monthly cost
18.42 USD

Total 12 months cost
221.04 USD
Includes upfront cost

**My Estimate**

| Service Name | Status | Upfront cost | Monthly cost | Description | Region | Config Summary |
|---|---|---|---|---|---|---|
| Amazon CloudWatch | - | 0.00 USD | 0.60 USD | Cloudwatch | US East (N. Virginia) | Number of Metrics (includes detailed and custom metrics) (2), Numbe… |
| Amazon EC2 | - | 0.00 USD | 8.52 USD | Bastion | US East (N. Virginia) | Tenancy (Shared Instances), Operating system (Windows Server), Wor… |
| Amazon EC2 | - | 0.00 USD | 9.30 USD | ASC | US East (N. Virginia) | Tenancy (Shared Instances), Operating system (Linux), Workload (Mon… |

## Conclusions

Auto scaling technologies are such powerful tools that can seamlessly help with the everlasting hassle of downtimes in projects, however, it is in fact a double-edged sword, because this automatic process still is a set of computer tasks, and computers will *always* do whatever they're told to do; this can lead to unexpected charges of (apparently) unrequested deployments. The results of this can be strange logic behavior, budget draining, resource overallocation, and other damaging actions.

Like any other AWS service, auto scaling has to be treated with care in order to produce desired and optimal results, and this can only be achieved through practices like this, in which non-real cases are developed, and hopefully the experience acquired from it can be useful in the future, in way more significant projects.

## Bibliography

[1] J. Barr. 'New Features for Amazon EC2: Elastic Load Balancing, Auto Scaling, and Amazon CloudWatch'. [Online]. Available: https://aws.amazon.com/blogs/aws/new-aws-load-balancing-automatic-scaling-and-cloud-monitoring-services/.

[2] Aws.amazon.com, 'Auto Scaling groups'. [Online]. Available: https://docs.aws.amazon.com/autoscaling/ec2/userguide/auto-scaling-groups.html.

[3] Aws.amazon.com, 'Amazon EC2'. [Online]. Available: https://aws.amazon.com/es/ec2/.

[4] nOps. 'Horizontal vs Vertical scaling: An in-depth Guide'. [Online]. Available: https://www.nops.io/blog/horizontal-vs-vertical-scaling/.