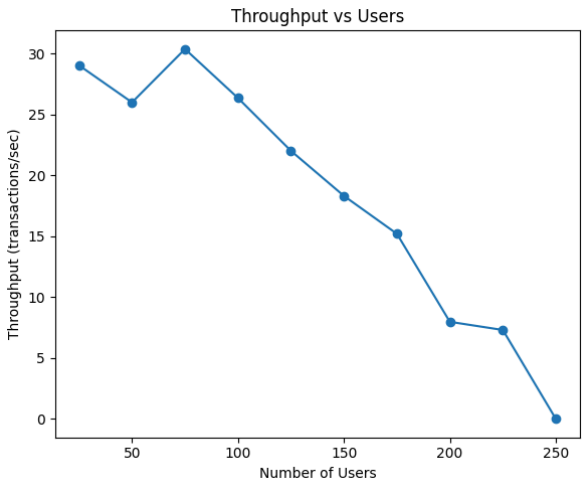
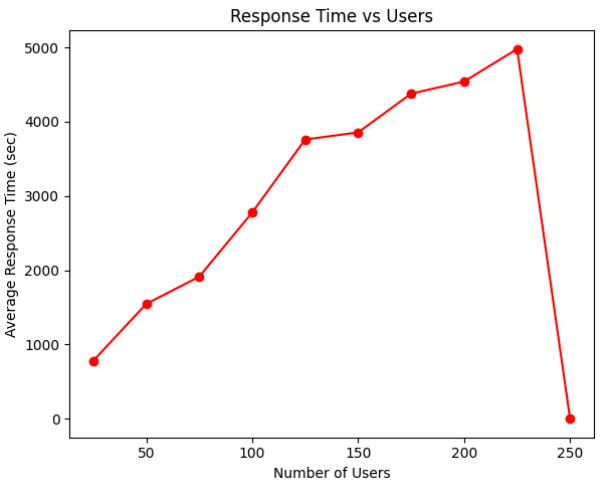
Chatrin Yoonchalard 6631304721

PaaS Auto Scaling

Baseline performance of PaaS





1. From part D, show the inflection point on both of your curves and explain how you map this to the appropriate system resources (performance metrics) to use to trigger auto-scaling for PaaS.

Ans: The inflection point is when the number of concurrent users is 75 users the throughput starts to drop. At that time the CPU Utilization is around 35% so I set the Upper threshold at 30% and lower at 20%

---

2. Describe the scaling policy you configured to scale up and scale down.

Ans: I configured Scaling cooldown to 60 second, Metric is CPU Utilization since our web app is matrix calculation which use CPU, Statistic to Maximum, Breach duration to 1 Min, Upper threshold 30%, and Lower threshold 20% this make it's very sensitive when our instance CPU Util reach the threshold 30% for 1 minute it's will trigger scaling operation.

Capacity

Environment type Load balanced	**Min instances** 1	**Max instances** 4
Fleet composition On-Demand instances	**On-demand base** 0	**On-demand above base** 0
Capacity rebalancing Disabled	**Scaling cooldown** 60	**Processor type** x86_64
Instance types t3.micro	**AMI ID** ami-0736ae53e38c421d9	**Availability Zones** Any
Metric CPUUtilization	**Statistic** Maximum	**Unit** Percent
Period 1	**Breach duration** 1	**Upper threshold** 30
Scale up increment 1	**Lower threshold** 20	**Scale down increment** -1

---

3. How did you run siege (what options) to trigger scaling up and scaling down?

Ans: I ran siege -c250 -t999H -b http://act-3-php-env.eba-tjaqjb2j.us-east-1.elasticbeanstalk.com/

This with 250 concurrent users to allow CPU Utilization to reach above 30% and time for 999 hours or occur until i interrupt execution this is to make sure the CPU utilization will be above 30% for more than 1 minute.

---

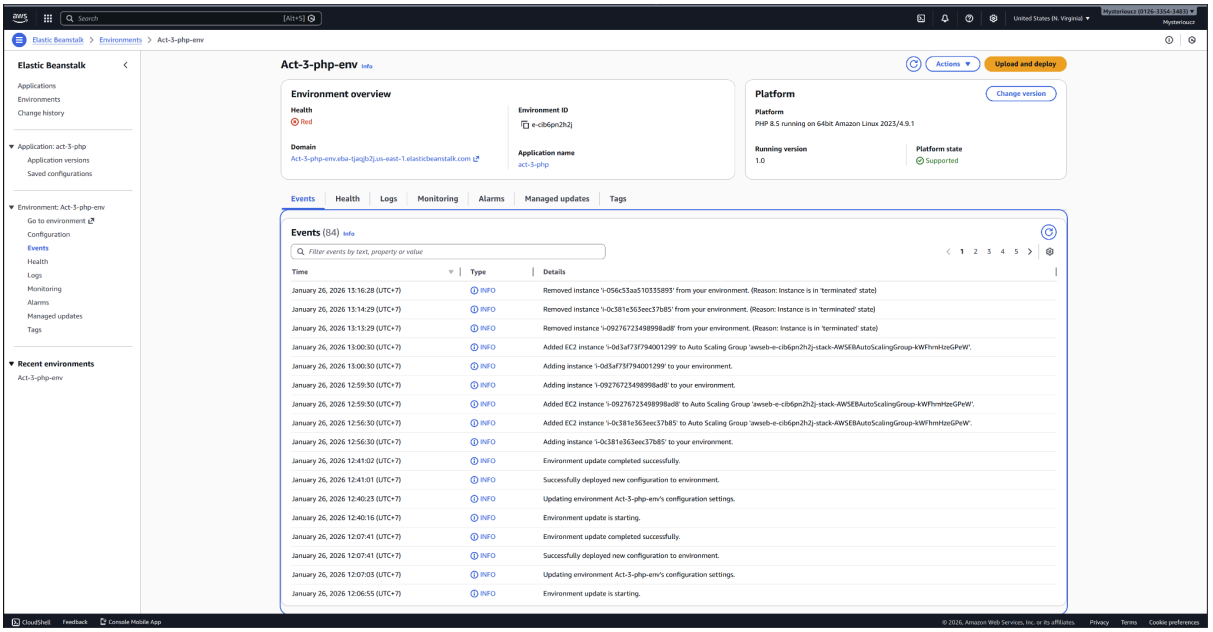
4. Was the scaling up and scaling down behavior consistent with your scaling policy?

Ans: Yes, the scale up and down is consistent with my policy. Scale up when CPU Utilization reaches 30% and Scale down when it is below 20%.

---

5. Include screenshots from Elastic Beanstalk showing the number of instances running and screenshots of resource monitoring data for your environment in your report to confirm that you successfully triggered auto-scaling according to your configured policies.

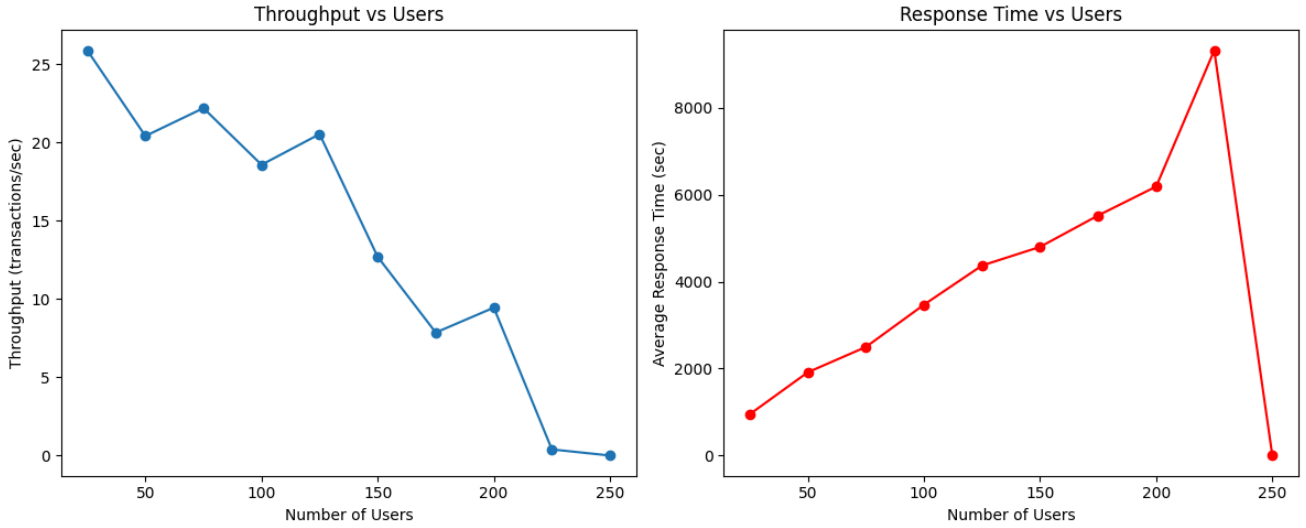
Ans: Here is the screen shot of the Event log which shows the ec2 instance added and removed.



Chatrin Yoonchalard 6631304721

IaaS Auto Scaling

Baseline Performance of IaaS



1. Consider observed baseline performance in terms of response time and throughput as a function of offered load (# of client requests/second) when you deploy the same web app to PaaS vs. IaaS. Double-check to see if they are deployed on the same actual instance sizes and check the AZ.

1.1. Discuss the differences in baseline performance in terms of response time and throughput as a function of offered load (# of client requests/second), and conclude which provides better performance?

> **Ans:** From the baseline performance results, PaaS (Elastic Beanstalk) performed better than IaaS. The average response time for PaaS was significantly lower than the manual IaaS setup under the same load conditions.

1.2. Are there any reasons that you think would sensibly explain the performance differences?

> **Ans:** I think it's because PaaS(Elastic Beanstalk) is using Nginx as a reverse proxy and has OPCache which results in Elastic Beanstalk processes the request and returns the HTML faster than untuned Apache server(IaaS).

---

2. For IaaS, what scaling policy did you use to scale up and scale down?

> **Ans:**
>
> - Scale Up: Triggered when Maximum CPU Utilization is above 70% for 30 seconds.
> - Scale Down: Triggered when Maximum CPU Utilization is below 30% for 30 seconds.

---

3. How did you run siege (what options) to trigger scaling up and scaling down?

Ans: For scale up I use siege -c 250 -t 30M -b http://phpiaasgroup-1-407212990.us-east-1.elb.amazonaws.com/.

This trigger scale up because it increase CPUUtilization to be above 70% for more than 30 seconds and for scale down I use siege -c 20 -t 3M -v http://phpiaasgroup-1-407212990.us-east-1.elb.amazonaws.com/.

This trigger scale down because it didn't use that much CPUUtilization(below 30%) for 30 seconds

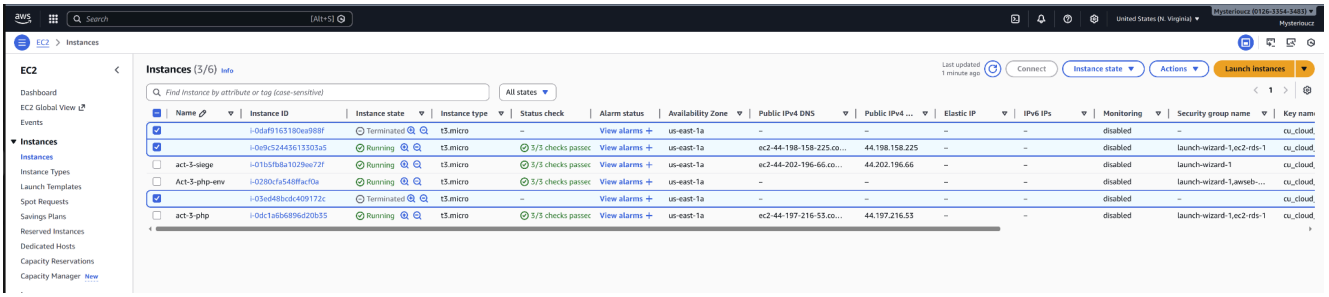
---

4. Was the scaling up and scaling down behavior consistent with your scaling policy?

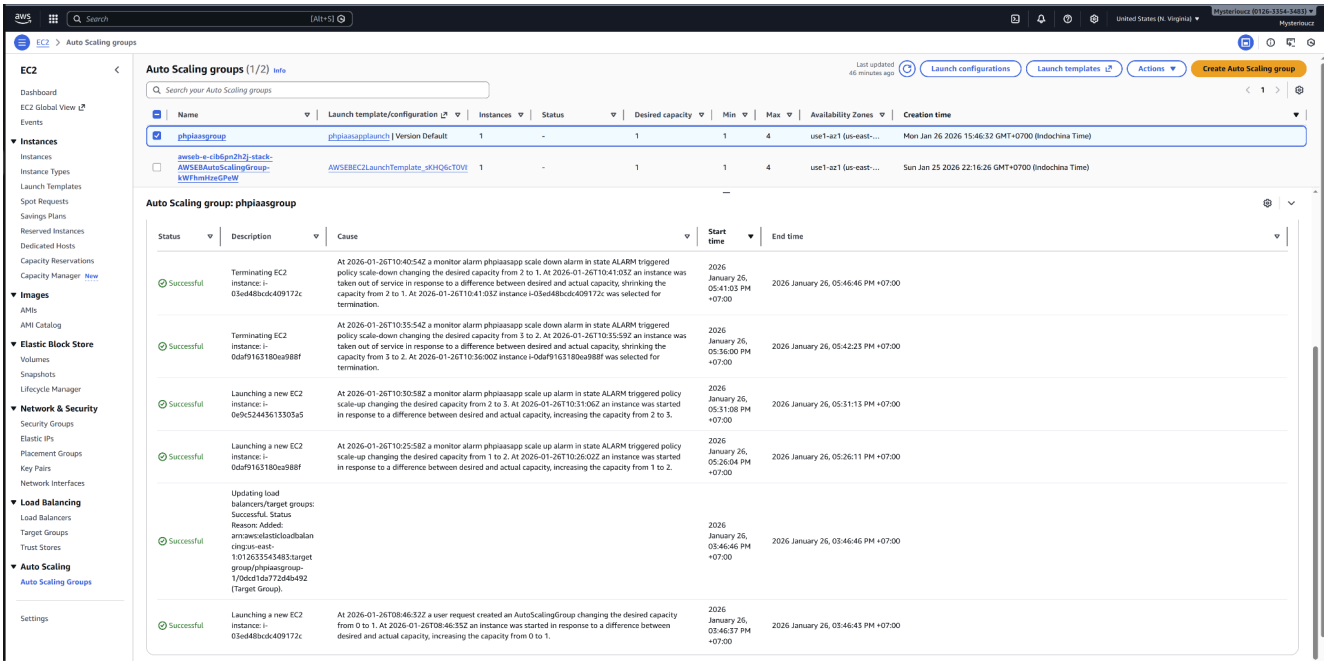
Ans: Yes, the behavior was consistent. The activity logs show new instances launching shortly after the CPU alarm was triggered, and instances being terminated when the load decreased.

---

5. Include screenshots from EC2 monitoring data in the report to make sure I can see that you did trigger auto-scaling according to your configured policies.



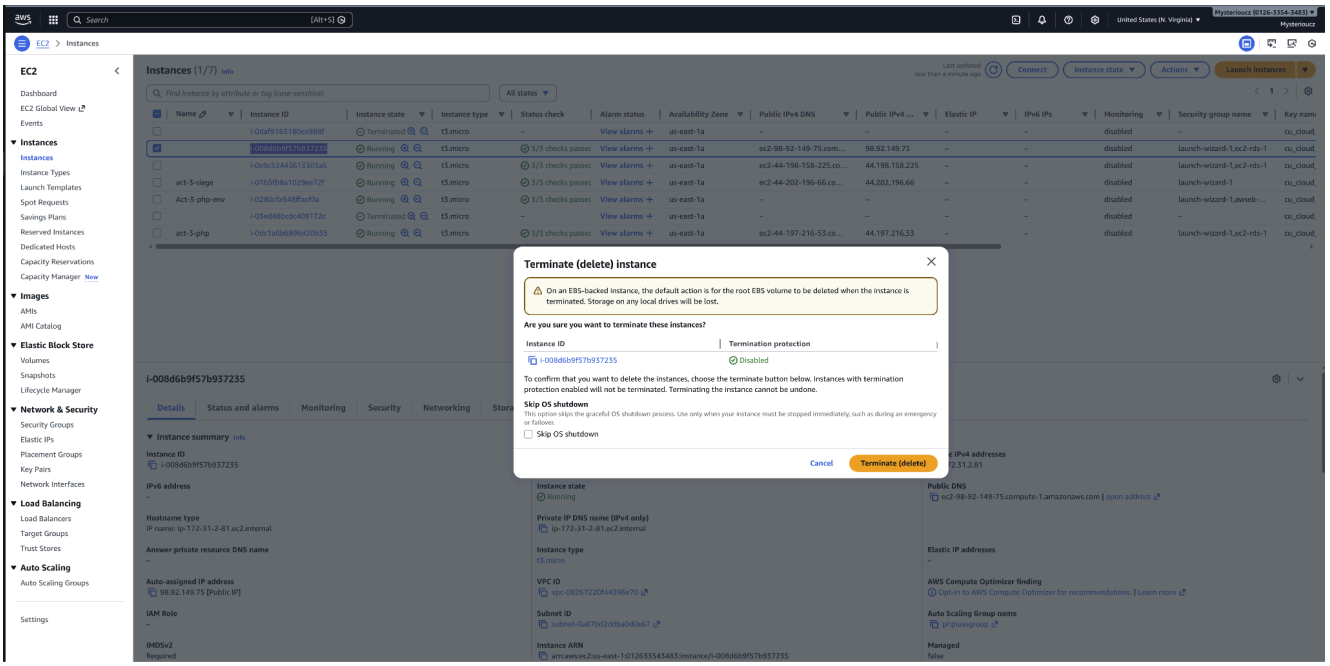
(3 selected instances are managed by AutoScalingGroup named "phpiaasgroup")



(Activity logs of Auto Scaling group "phpiaasgroup")

---

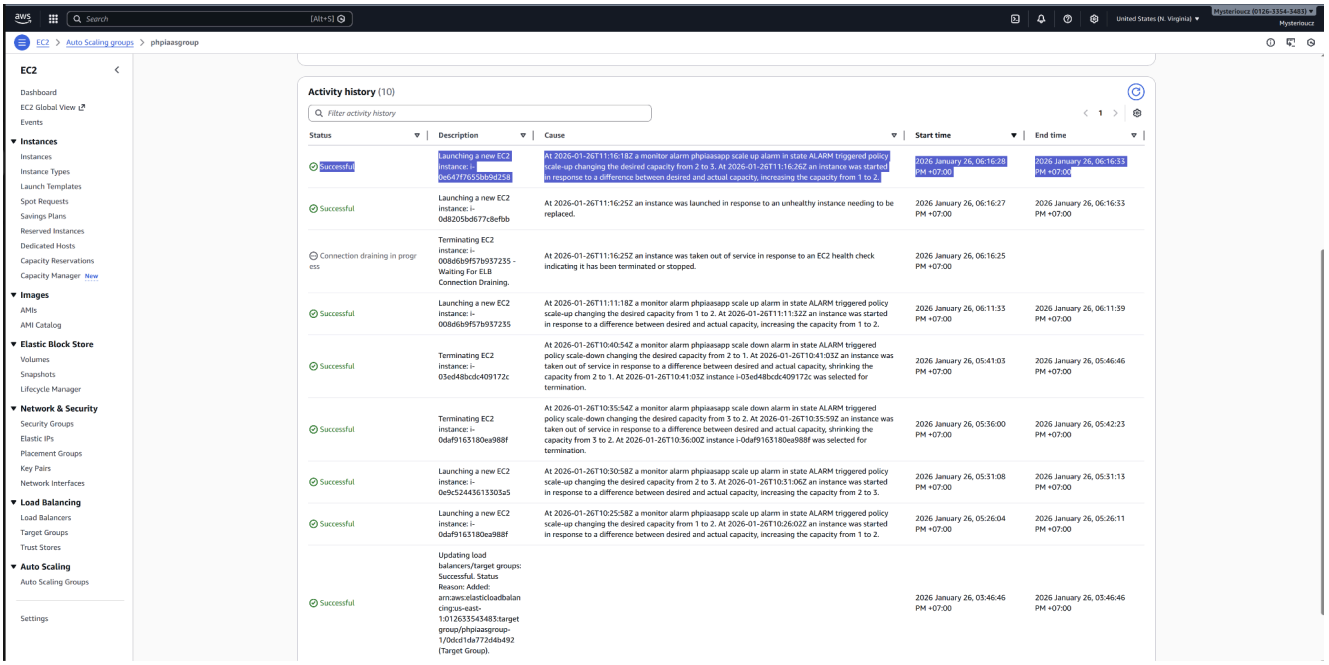
6. Consider what Amazon did when you terminated your EC2 instance(s) in part D (fault tolerance).



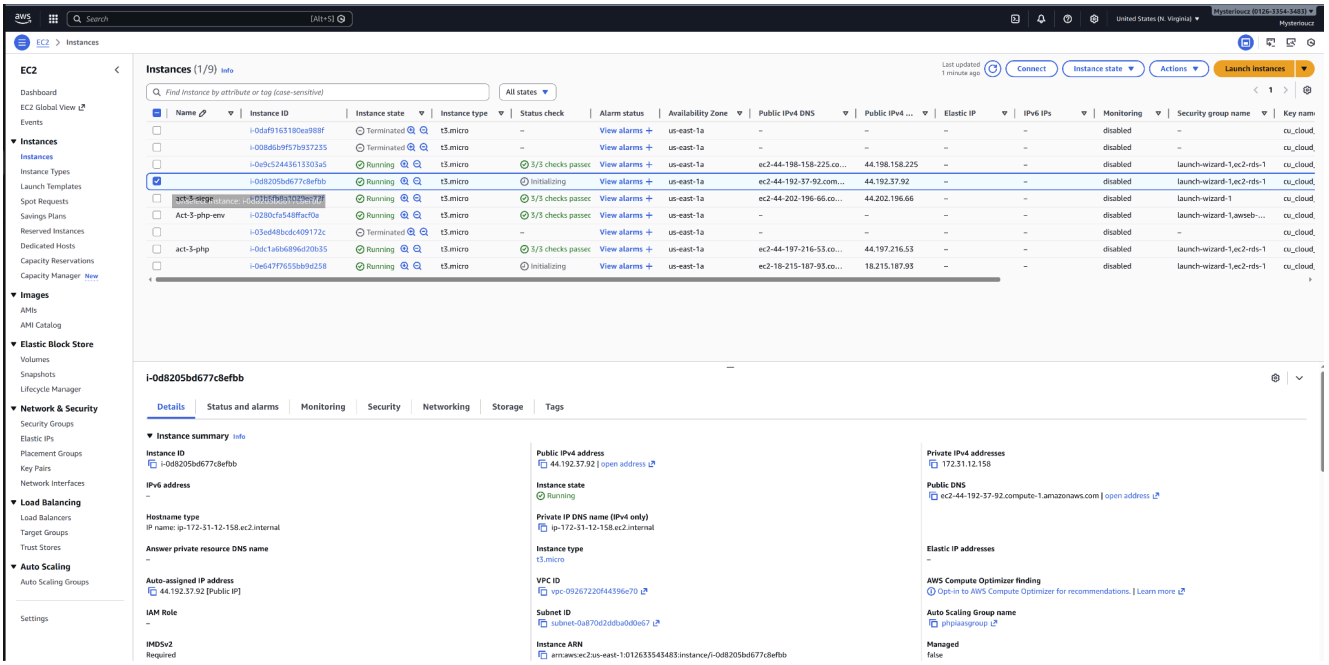
(terminated an instance)

6.1. Is Amazon able to launch new instances to replace the one(s) you killed?

Ans: Yes. The Auto Scaling Group detected the unhealthy (terminated) instance and automatically launched a new replacement instance to maintain the desired capacity.



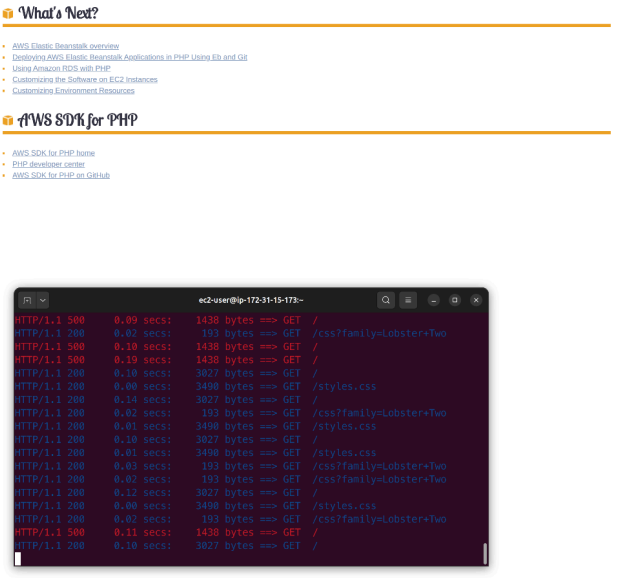
(Activity history that Auto Scaling group launching new instances 2 upper rows)



(The lowest and selected row is the latest instances that got launched)

6.2. When the new instances are launched, are your web clients (browser or siege) able to access the web application like normal? Provide evidence of what your web clients see.

Ans: Yes. Once the new instance passed its health checks, the Load Balancer resumed sending traffic to it. My Siege client and browser could access the web application normally without manual intervention.



---

7. Discuss the differences between configuring auto-scaling in IaaS vs. auto-scaling PaaS. What are the strengths of each configuration approach?

Ans: Both can achieve the same configuration but there are some differences.

- **PaaS (Elastic Beanstalk):** Configuration is much simpler. All settings (scaling triggers, load balancing, health checks) are centralized on a single "Configuration" dashboard.
- **IaaS (Manual):** Requires more manual steps. I had to separately configure the Launch Template, Target Groups, Load Balancer, and Auto Scaling Group policies.

---

8. Discuss the differences between IaaS vs. PaaS. What are the strengths of each approach?

Ans: I think both have trade offs.

- **IaaS Strengths:** Offers higher flexibility and control. We can customize the OS, install specific kernel modules, and tune the network stack exactly how we need it.
- **PaaS Strengths:** Offers higher convenience and speed. It handles deployment, patching, and scaling logic automatically, allowing developers to focus on code rather than infrastructure.
