Roselle Paala

CIS 6930/Cloud Computing

Dr. Phillip Ventura

April 7, 2019

**AWS AUTO SCALING**

Generally, when the topic of AWS Auto Scaling comes up, the concept of Fleet Management follows. To better understand Auto Scaling, let’s first define Fleet Management. An Application that runs on Amazon EC2 instances is often referred to as "fleet". An Amazon EC2 Fleet is a new feature that simplifies the planning of an Amazon EC2 capacity across its various features. These features could be the Amazon EC2 instance types, Availability Zones and across On-Demand, Amazon EC2 Reserved Instances (RI), and Amazon EC2 Spot purchase models.

Amazon introduced Auto Scaling to simplify Fleet Management. There are three main functions that Auto Scaling allows for:

* Monitors the health of running instances
* Automatic replacement of impaired instances
* Balances the capacity across Availability Zones

**Monitors the Health of Running Instances**

AWS Auto Scaling monitors the health of all instances placed in the Auto Scaling group. It performs EC2 health checks and ELB health checks (if the instance is connected to an Elastic Load Balancing load balancer) regularly. Auto Scaling makes sure that the application on incoming traffic is allowed and that an instance is functioning properly. If a failure in health checks is detected, Auto Scaling replaces the instance automatically.

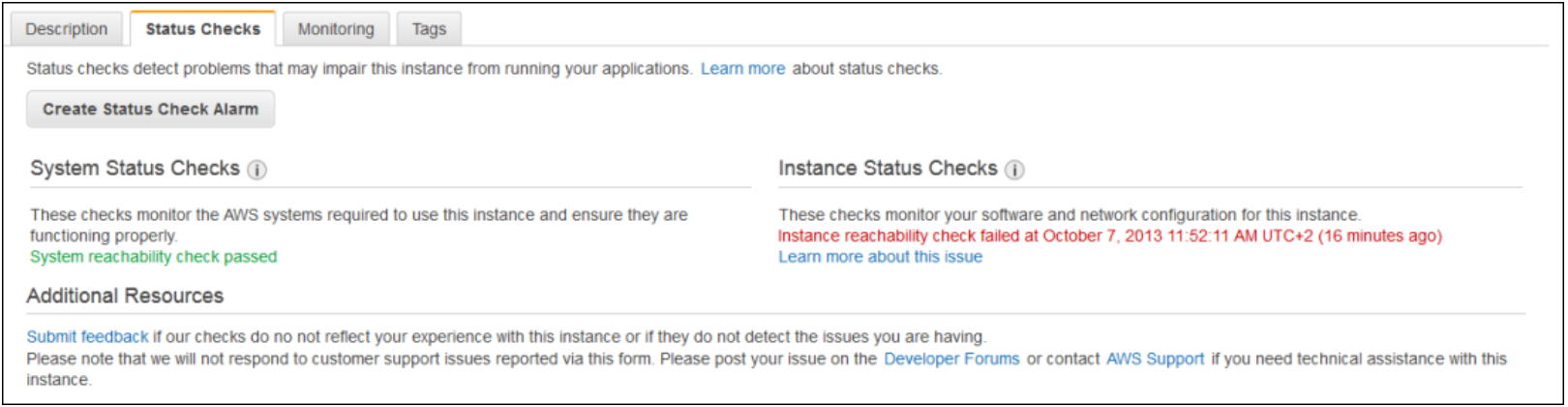


Fig 1: Example of System Status Checks

**Auto Scaling Automatically Replaces Impaired Instances**

Following EC2 health checks, when an impaired instance is detected, Auto Scaling terminates and replaces the instance with a new one automatically. If an Elastic Load Balancing load balancer is used, the impaired instance is detached from the load balancer before a new instance is created and attached back to the load balancer.

**Auto Scaling Balances Capacity Across Availability Zones**

One of Auto Scaling’s most important function is its ability to balance the capacity of Availability Zones. Auto Scaling ensures that EC2 instances are distributed evenly across Availability Zones and applications are well-architected. Auto Scaling automatically creates instances in Availability Zones. It only creates the instance into an Availability Zone that has the available capacity for this requested instance type.

**Implement Auto Scaling in Your Instances**

Auto Scaling is very easy to implement if you have an existing instance. However, before creating your Auto Scaling group, consider the following:

* How long does your application take to launch and configure instances
* What metrics are most important for your application to perform optimally
* The number of Availability Zones the Auto Scaling group should span
* Available resources such as security groups and AMIs (Amazon Machine Images).
* Your purpose for creating an Auto Scaling group (ie. Is your desire to increase or decrease capacity?)

Once considerations are identified, you are ready to get started. First, ensure that you are logged in to your AWS Management Console. If this is your first time creating an Auto Scaling group, you will have to Create a Launch Template (Fig. 2). Then, create an Auto Scaling group using a launch template (Fig. 3 or Fig. 4). View the currently running instance(s). Then, right click on the running instance and choose **Instance Settings** (Fig.2) **--> Attach to Auto Scaling Group.** If this is your first time attaching the instance to Auto Scaling Group, you have the option to create a new Auto Scaling Group. For illustrated steps on creating a new Auto Scaling Group, see Fig. 3 – Fig. 11. Once the instance(s) is attached to the Auto Scaling Group, your instance(s) health will be automatically monitored and when errors or failures are detected, the instance will be automatically replaced with a new instance. As you add additional zones and create new instances, they will be spread throughout the Availability Zones.

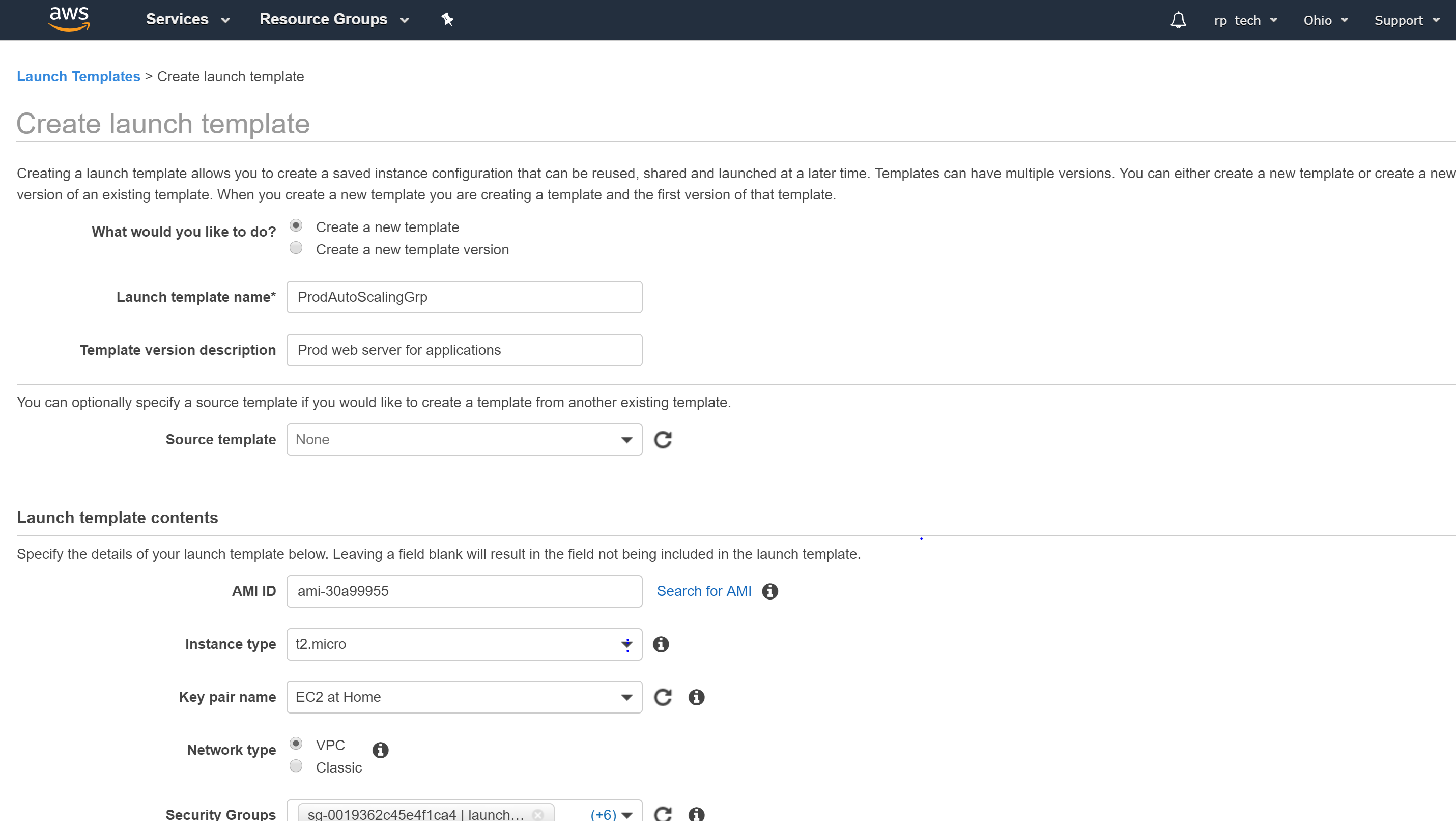


Fig. 2

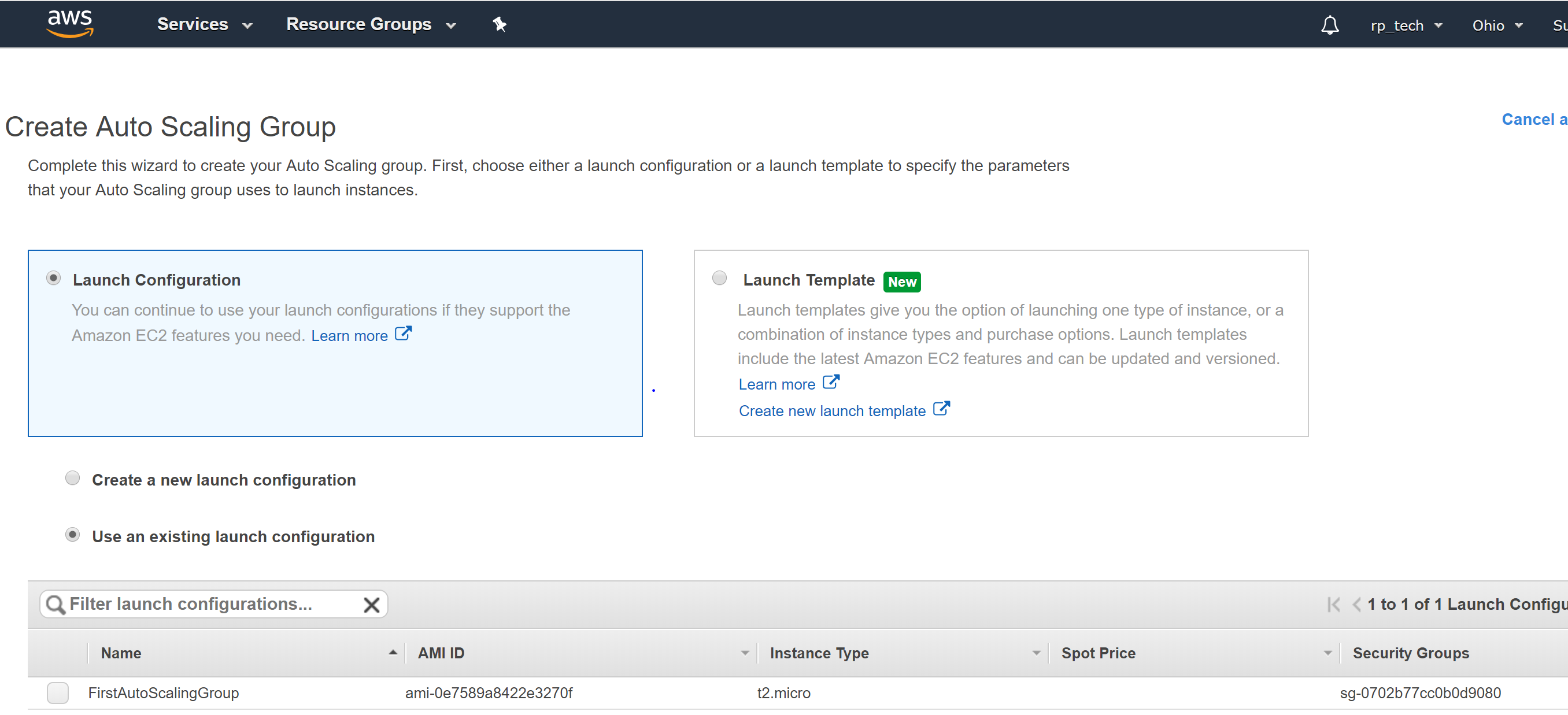


Fig. 3

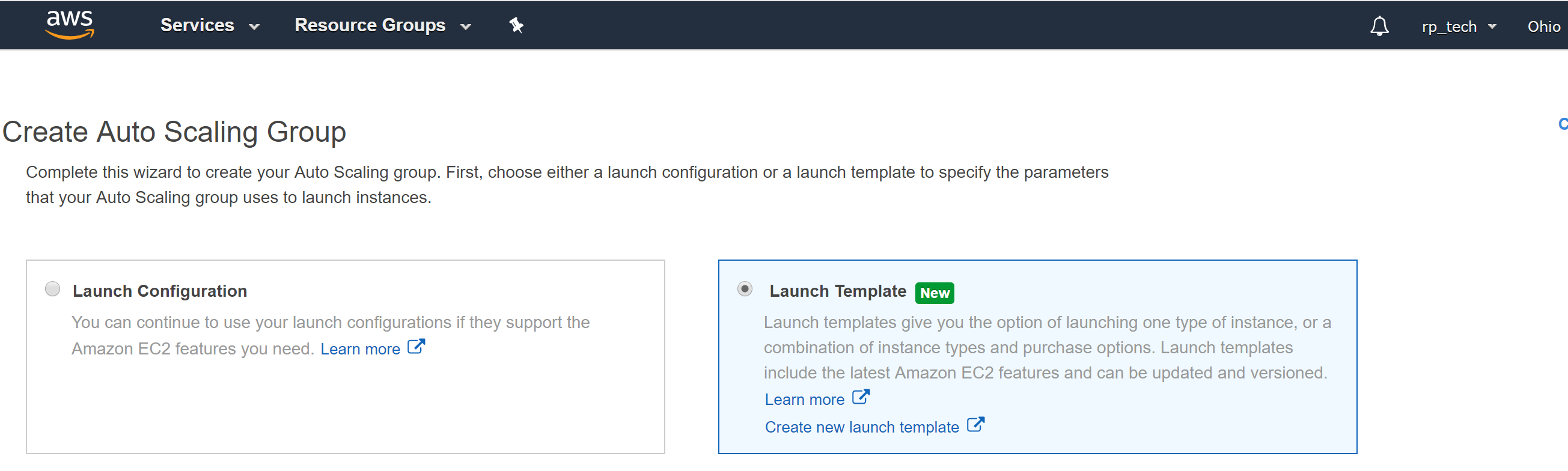


Fig. 4

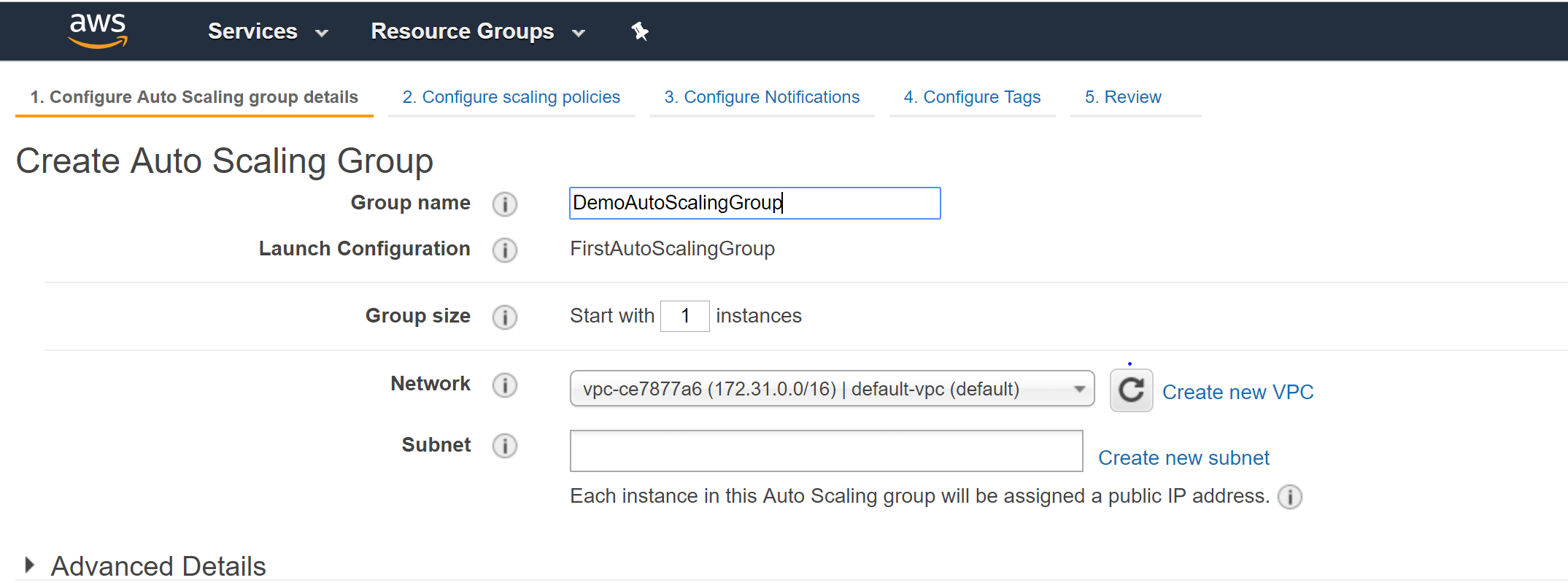
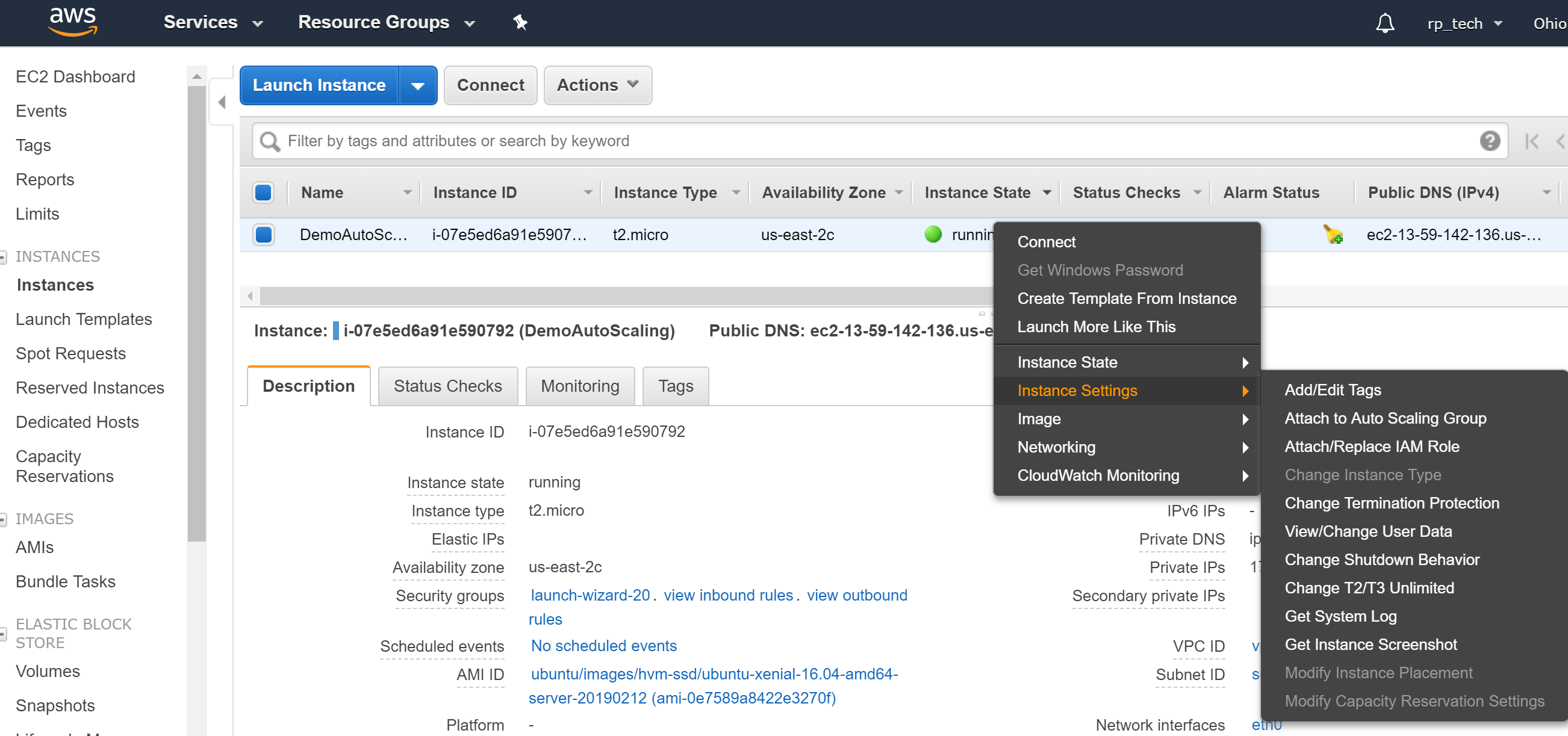
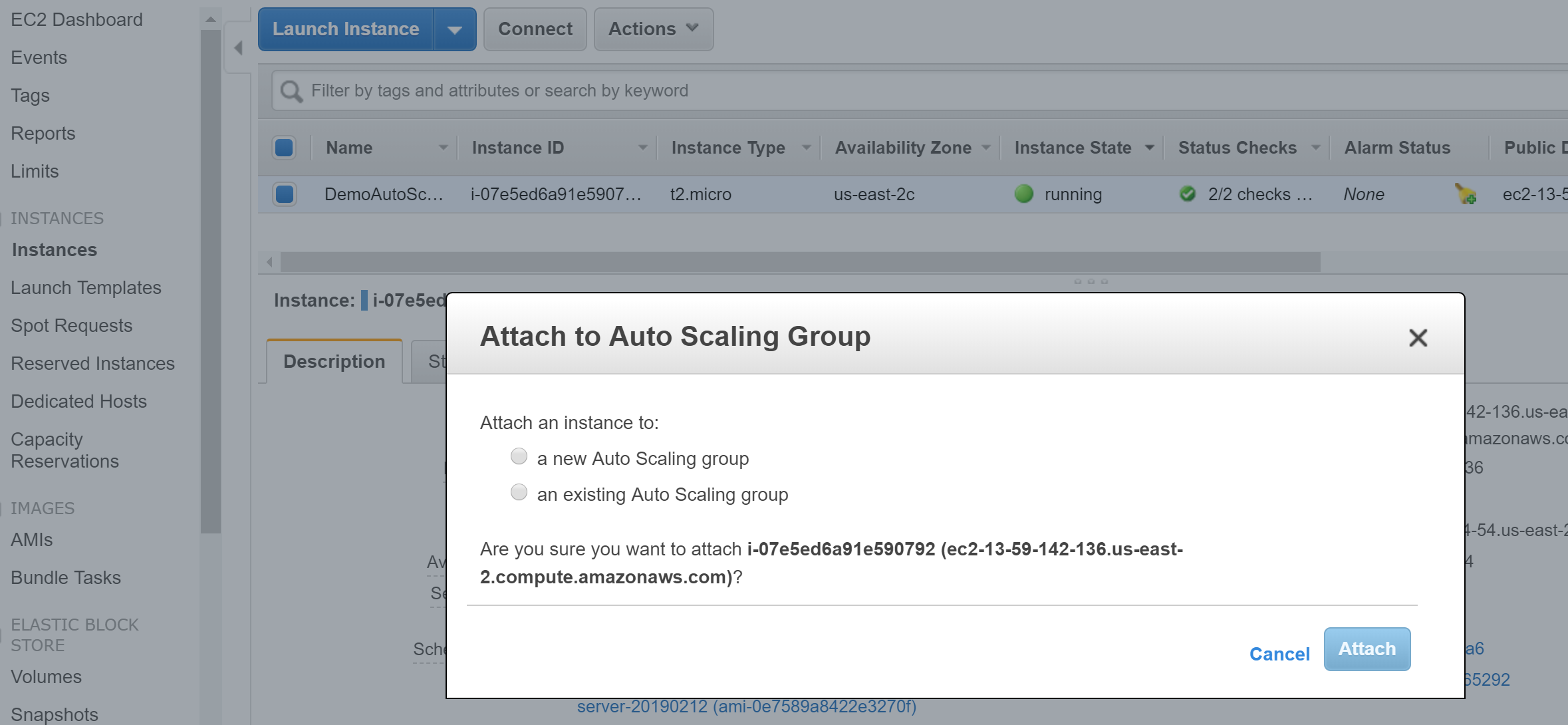


Fig. 5





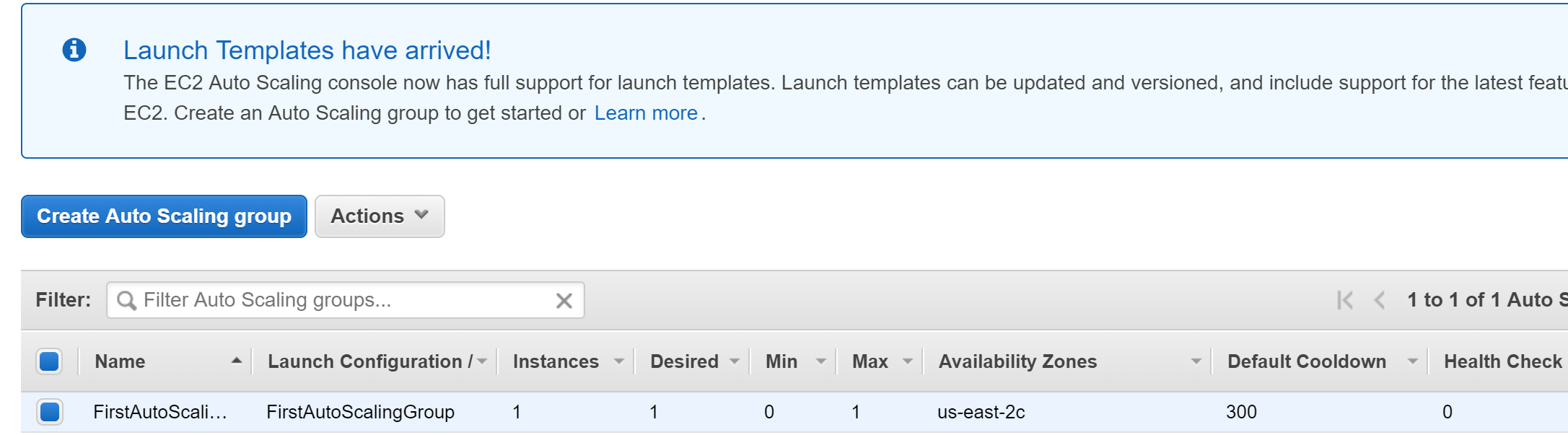
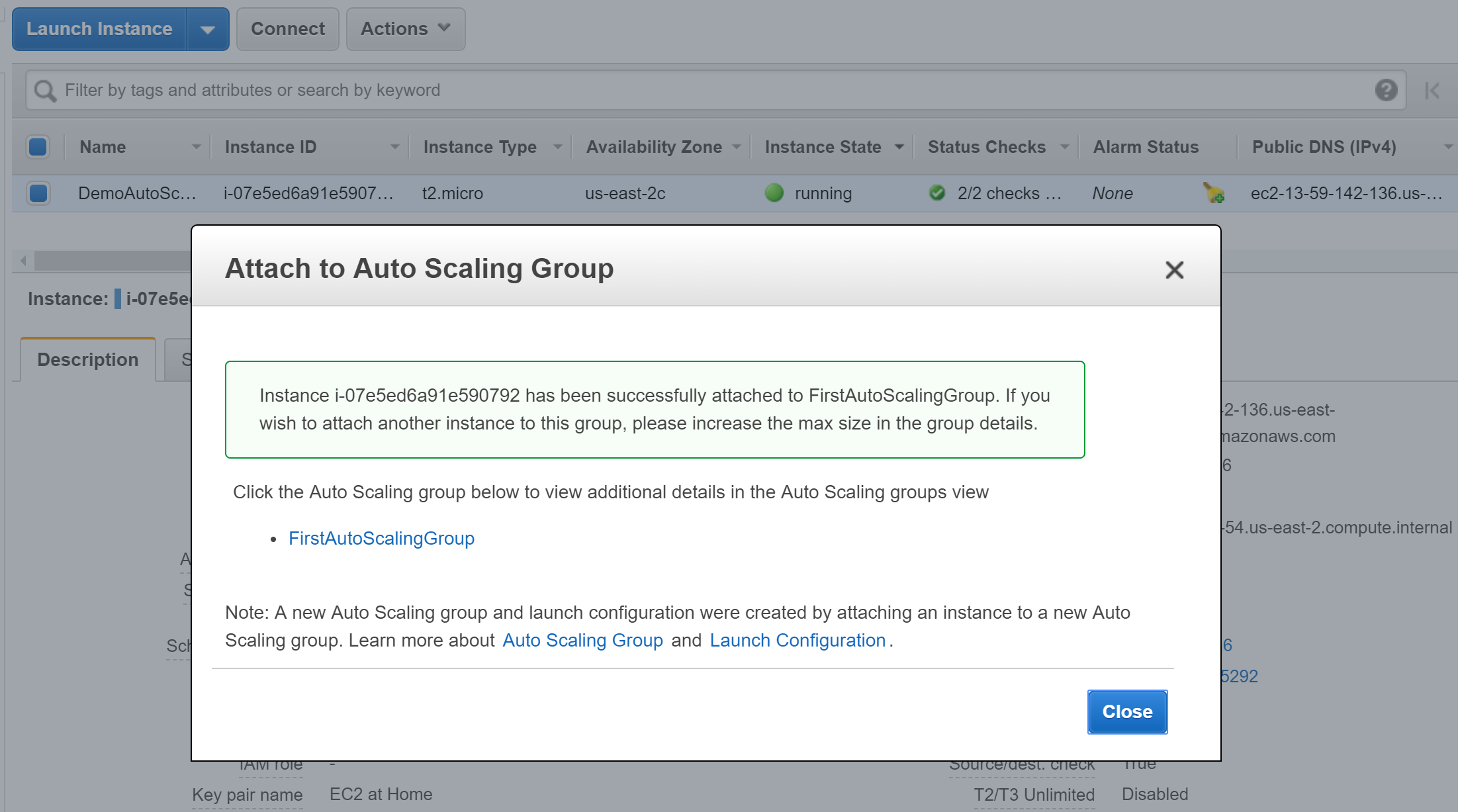


Fig. 4

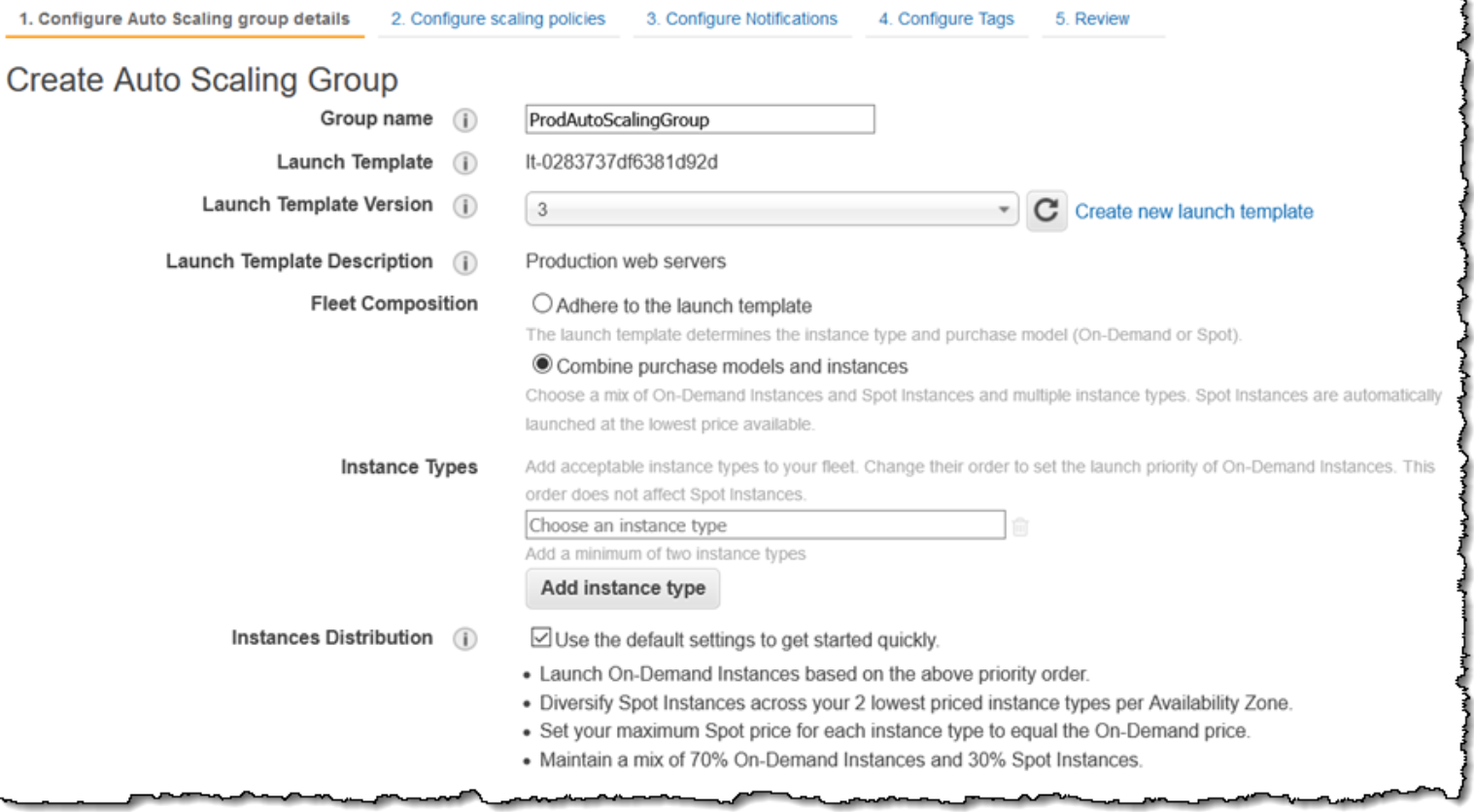


Fig. 5

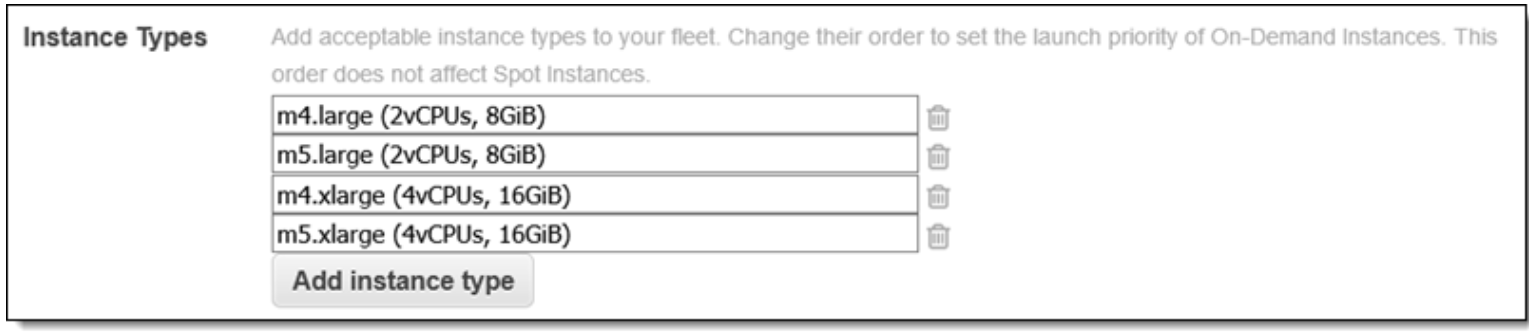


Fig. 6

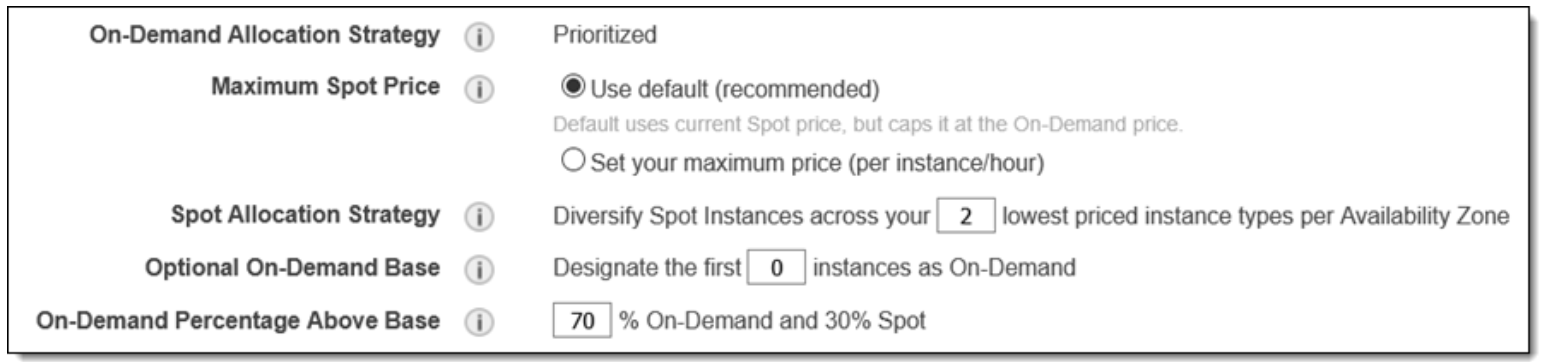


Fig. 7

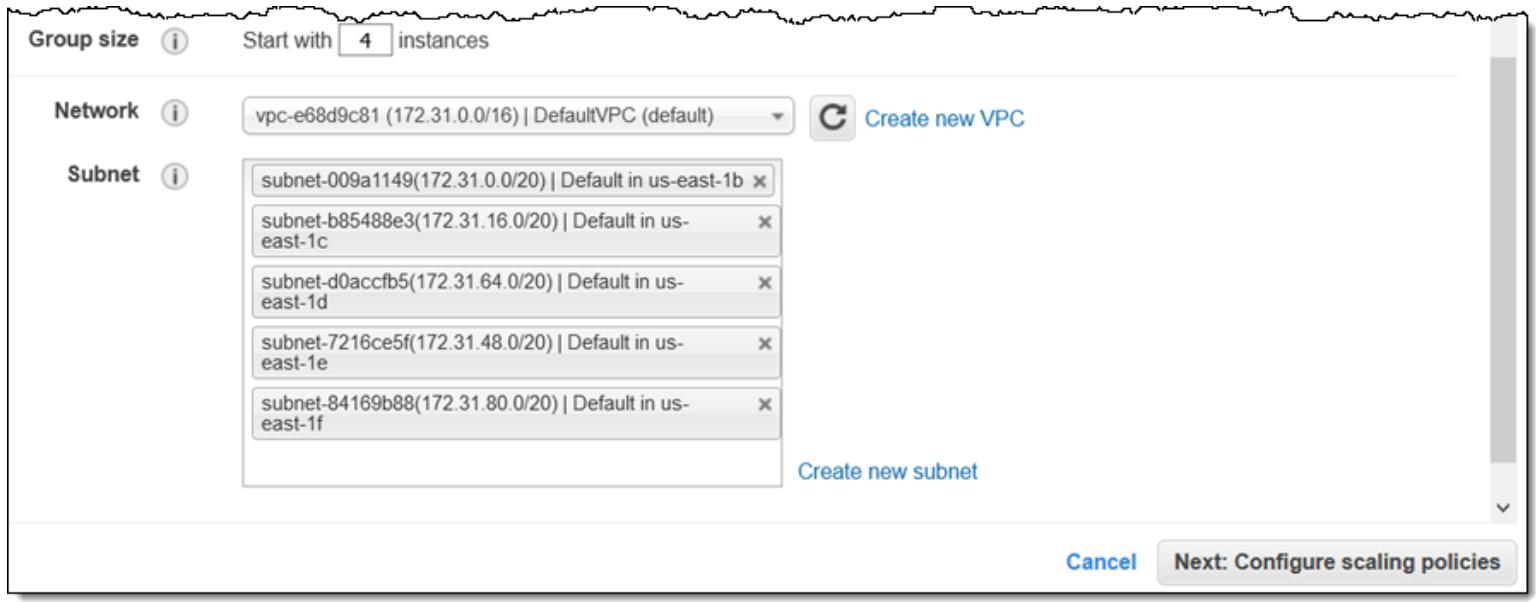


Fig. 8

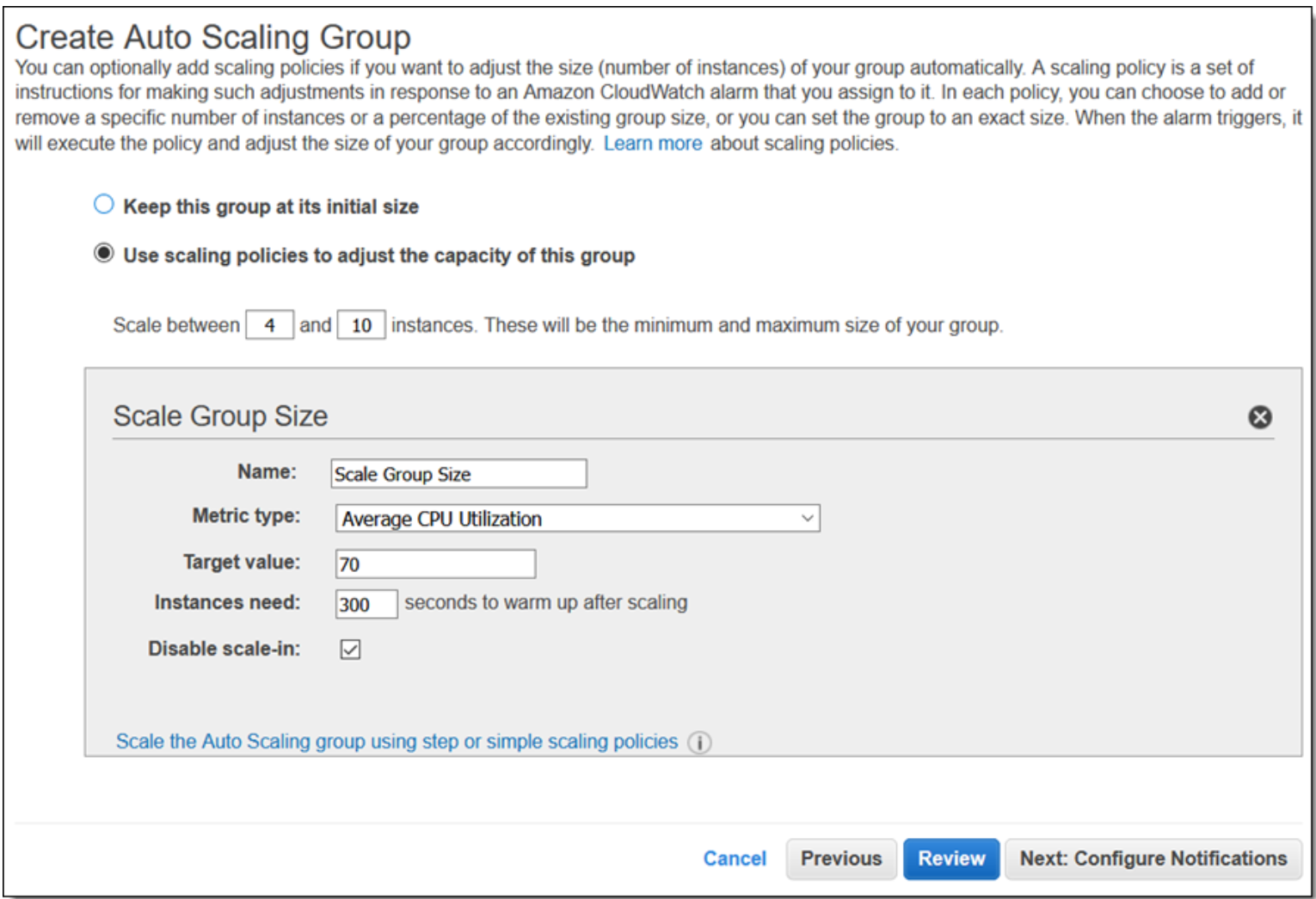


Fig. 9

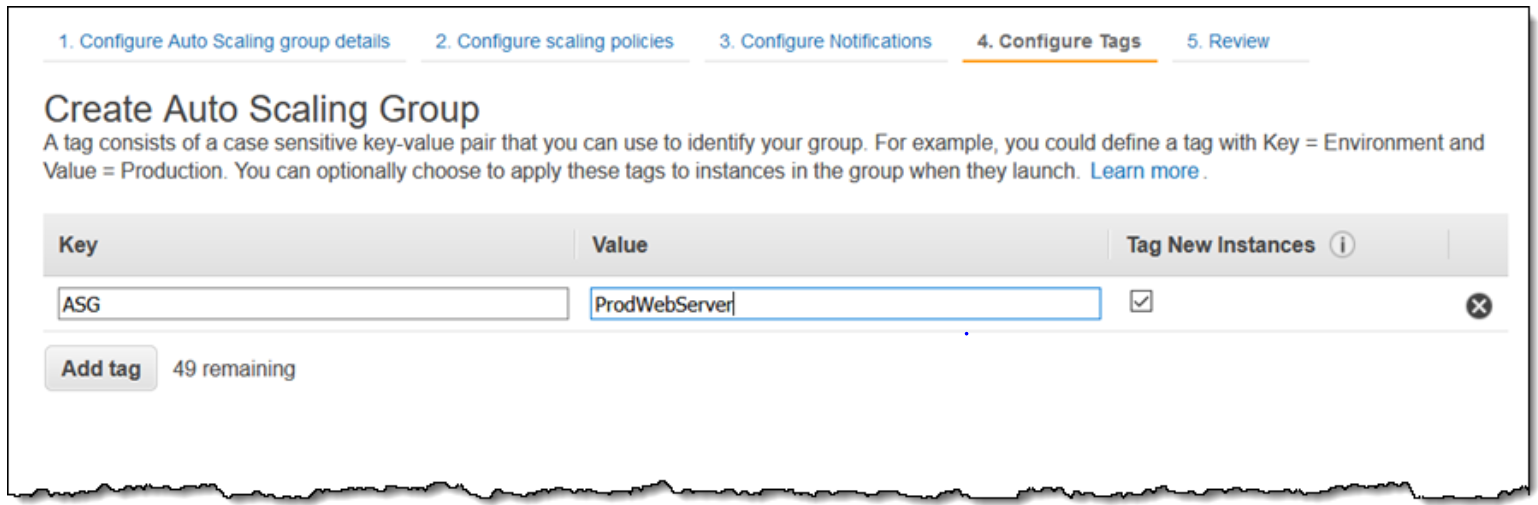


Fig. 10

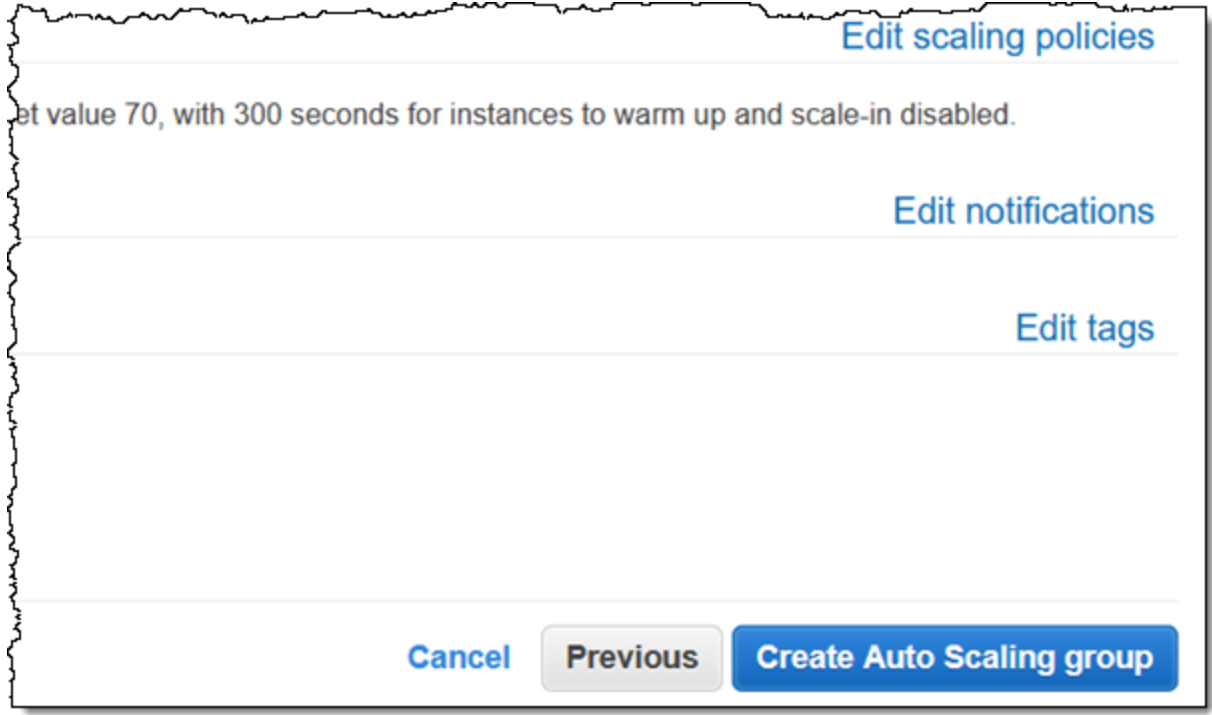


Fig. 11

**Extend Your Knowledge of Auto Scaling**

The previous information discussed is an introduction to Auto Scaling. Should you wish to further explore Auto Scaling, you can automate the management of your EC2 instance(s). Take for instance the use of AWS Beanstalk. [AWS Beanstalk](https://aws.amazon.com/elasticbeanstalk/) allows automatic software deployments, particularly for web applications. AWS [CodeDeploy](https://aws.amazon.com/codedeploy/) is another suggested solution. Additionally, consider [Amazon EC2 Containers Services](https://aws.amazon.com/ecs/) (Amazon ECS) if your application is container-based. Lastly, consider Blue/Green Deployments on fleet management. [Blue/Green Deployments](https://www.slideshare.net/AmazonWebServices/dvo401-deep-dive-into-bluegreen-deployments-on-aws/25) offers an effective strategy to avoid further downtime on your fleet.

**Additional Features of Auto Scaling**

**Launch Configurations, Lifecycle Hooks, and Fleet Size**

When Auto Scaling groups are launched, it uses the previously associated configurations. You may create your own launch configurations or, if you are using an Amazon Machine Image (AMI), you can point the launch configurations to the version you want Auto Scaling to deploy into your instances.

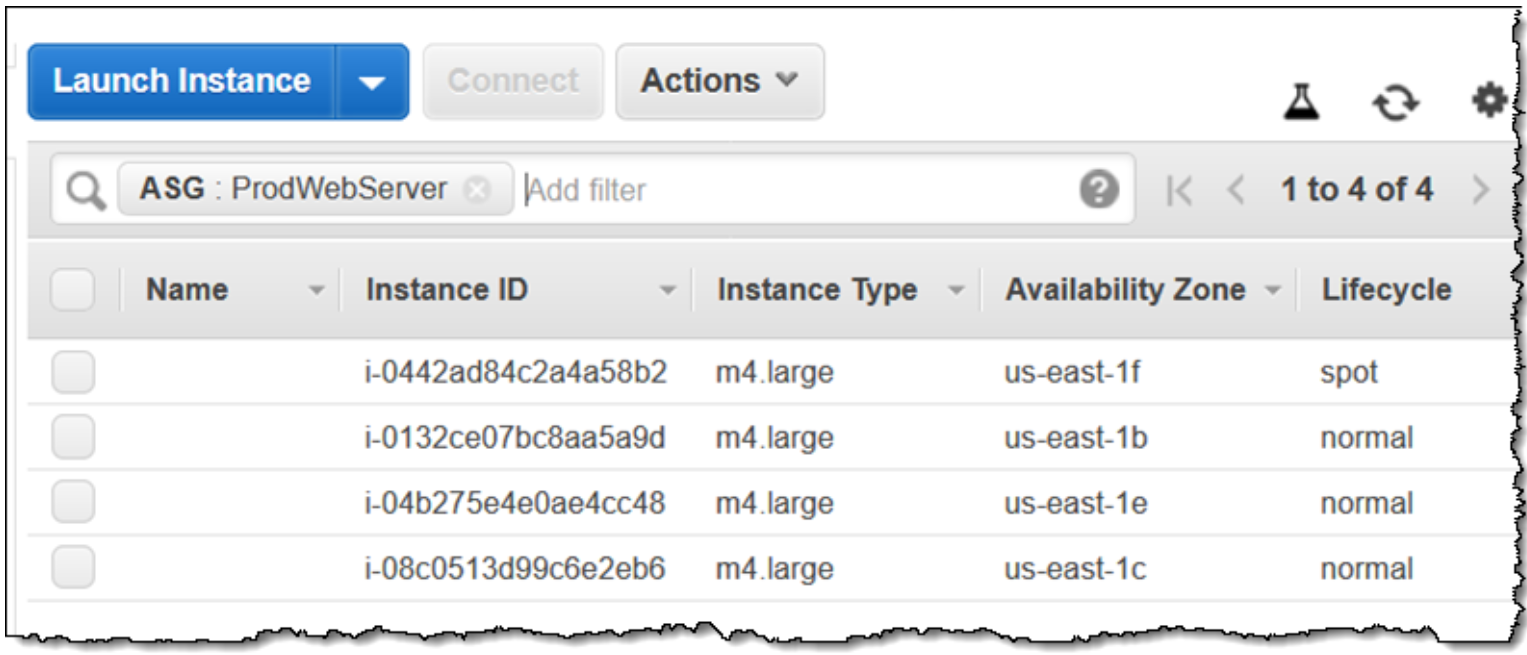
Lifecycle hooks allow for custom actions by temporarily interrupting running instances while an Auto Scaling group launches or terminates the instance. When an instance is in the idle state, it allows the user to install or configure the software and confirm that it is fully ready before incoming traffic is allowed.

Through Launch Configurations, you control the minimum, maximum, and desired attributes of an Auto Scaling Group. As previously mentioned, defined attributes are automated, and instances are distributed evenly throughout the Availability Zones.

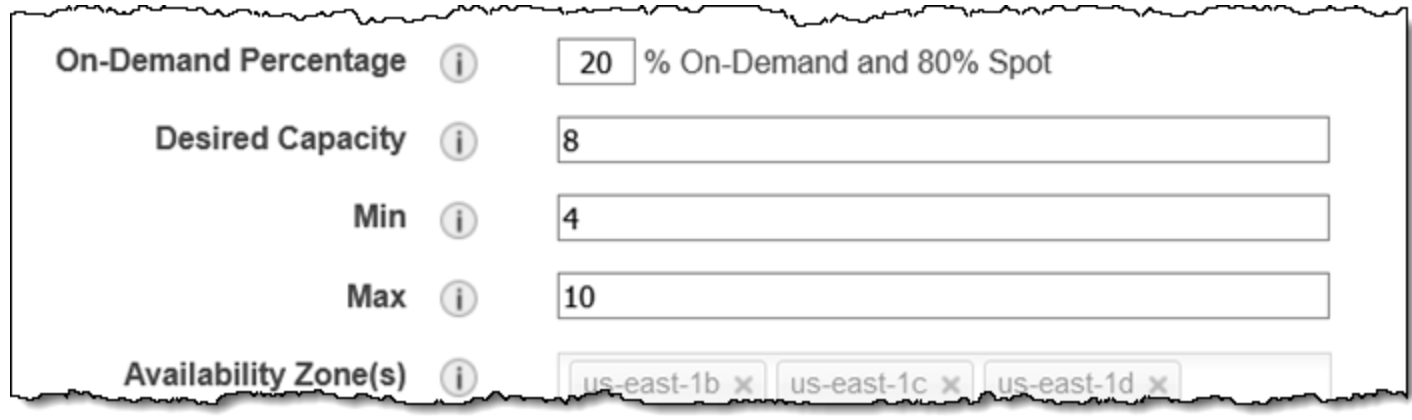
**Automatic Process Control and Scheduled Scaling**

The automated process can be modified in the desired behavior. For instance, you may disable Auto Scaling’s action of automatically terminating instances from one zone and re-launch into another instance. Furthermore, you may control when health checks, launches, and terminations are implemented.

Another great feature of Auto Scaling is the ability to schedule scaling. You may set up the time that is optimal for your application when traffic is less congested. [Dynamic Scaling](https://docs.aws.amazon.com/autoscaling/ec2/userguide/as-scale-based-on-demand.html) is offered as an advanced tool for scheduled scaling.

Filter the column to display the Lifecycle column:

Modify the Auto Scaling Group to reduce the On-Demand Percentage to 20% and double the Desired Capacity:



**Test Your Knowledge of Auto Scaling**

The purpose of this assignment is to apply the knowledge that you have learned from Amazon EC2 Auto Scaling. You must create an Auto Scaling Group on an existing instance. If you do not have running instances, try to create a desired amount of instance. Once implemented, try to apply one to two features that were introduced in the lecture.

References

**Website:**

Amazon EC2 Auto Scaling. (n.d.). Retrieved April 6, 2019, from https://aws.amazon.com/ec2/autoscaling/?hp=tile&so-exp=below

Amazon EC2 Auto Scaling Lifecycle Hooks. (n.d.). Retrieved April 6, 2019, from https://docs.aws.amazon.com/autoscaling/ec2/userguide/lifecycle-hooks.html

Barclay, C. (2016, October 20). Fleet Management Made Easy with Auto Scaling | Amazon Web Services. Retrieved April 6, 2019, from https://aws.amazon.com/blogs/compute/fleet-management-made-easy-with-auto-scaling/

Barr, J., & Barr, J. (2018, November 15). New – EC2 Auto Scaling Groups With Multiple Instance Types & Purchase Options | Amazon Web Services. Retrieved April 6, 2019, from https://aws.amazon.com/blogs/aws/new-ec2-auto-scaling-groups-with-multiple-instance-types-purchase-options/

Introducing Amazon EC2 Fleet. (n.d.). Retrieved April 6, 2019, from https://aws.amazon.com/about-aws/whats-new/2018/04/introducing-amazon-ec2-fleet/

Status Checks for Your Instances. (n.d.). Retrieved April 6, 2019, from <https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/monitoring-system-instance-status-check.html>

AWS Elastic Beanstalk – Deploy Web Applications. (n.d.). Retrieved April 6, 2019, from https://aws.amazon.com/elasticbeanstalk/