

**One of the biggest problems of Jenkins is its scalability. To achieve a decent result we run our builds on dedicated Jenkins slaves. These EC2 instances are running all the time. But this solution is not cost-effective. What could be a good approach to provision these slaves in AWS?**

ANS: One approach could be to deploy the slave instances using AWS Auto Scaling. In Auto Scaling Group we can define the AMI which should be used to launch the new instances and also we can define the number of min and max instances. The criteria to launch new instances could be a memory or CPU threshold. For eg: if the CPU threshold reaches 90% then launch a new instance. The extra instances will be terminated once the CPU usage becomes normal.

In this way, the new instances will only be launched when there is a requirement, otherwise the auto scaling group will only keep the minimum defined instances in the auto scaling group, this will help in minimizing the cost.