**Deploying Spring Boot Application on Elastic Beanstalk**

**Elastic Beanstalk Intro:**

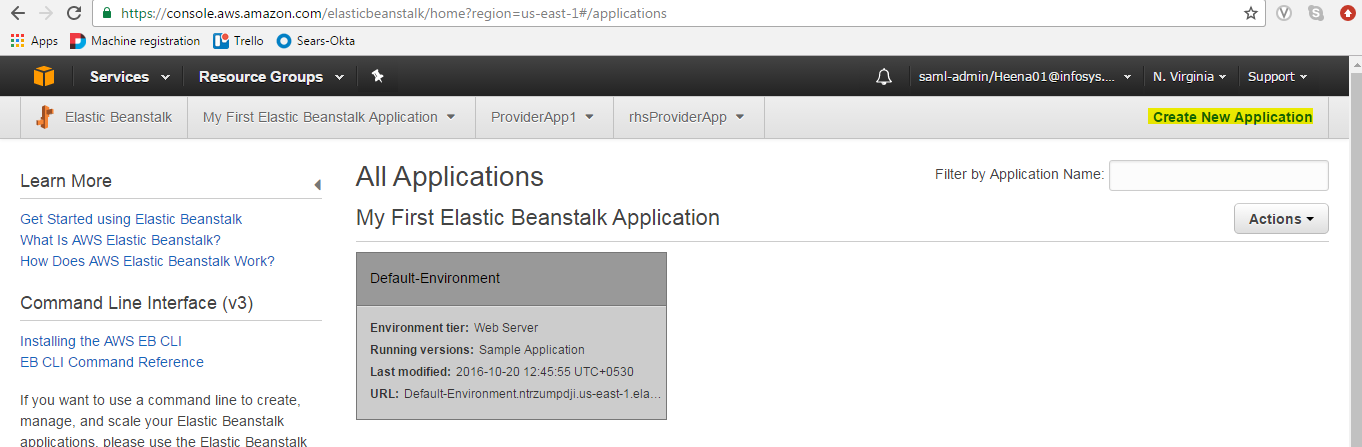
1. AWS Elastic Beanstalk is an easy-to-use service for deploying and scaling web applications and services developed with Java, [.NET](https://aws.amazon.com/net/), PHP, Node.js, Python, Ruby, Go, and [Docker](https://aws.amazon.com/docker/) on familiar servers such as Apache, Nginx, Passenger, and [IIS](https://aws.amazon.com/windows/).
2. You can simply upload your code and Elastic Beanstalk automatically handles the deployment, from capacity provisioning, load balancing, auto-scaling to application health monitoring. At the same time, you retain full control over the AWS resources powering your application and can access the underlying resources at any time.
3. There is no additional charge for Elastic Beanstalk - you pay only for the AWS resources needed to store and run your applications.

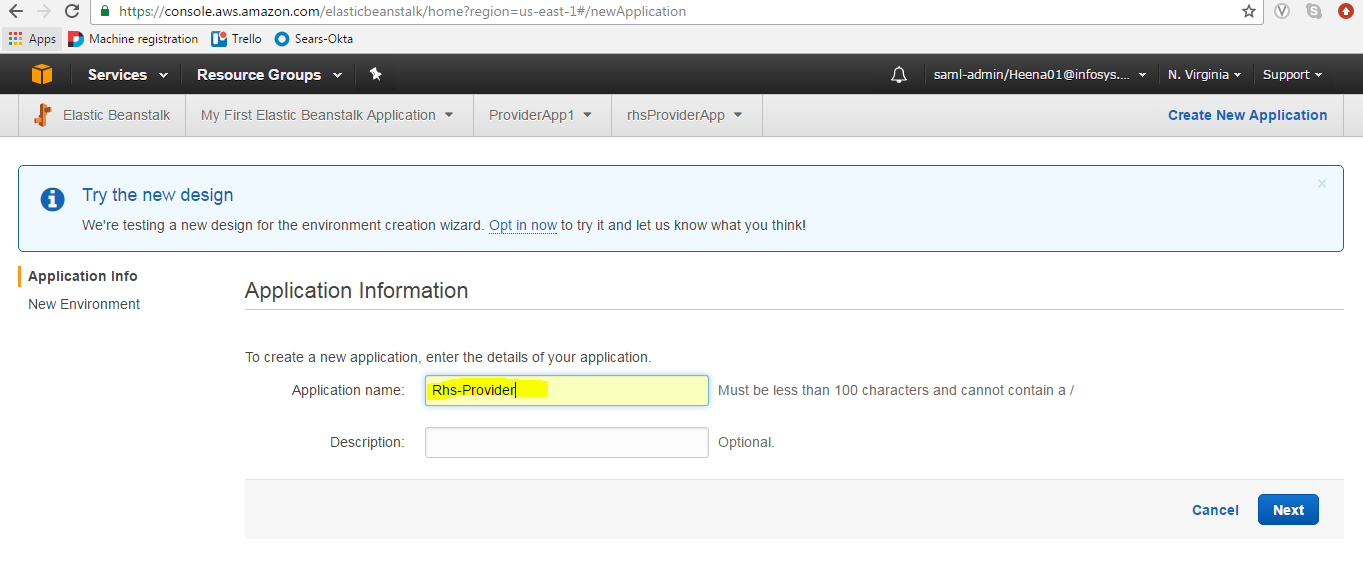
**Deploying Elastic Bean Stalk Application** (*rhs-provider Application here*):

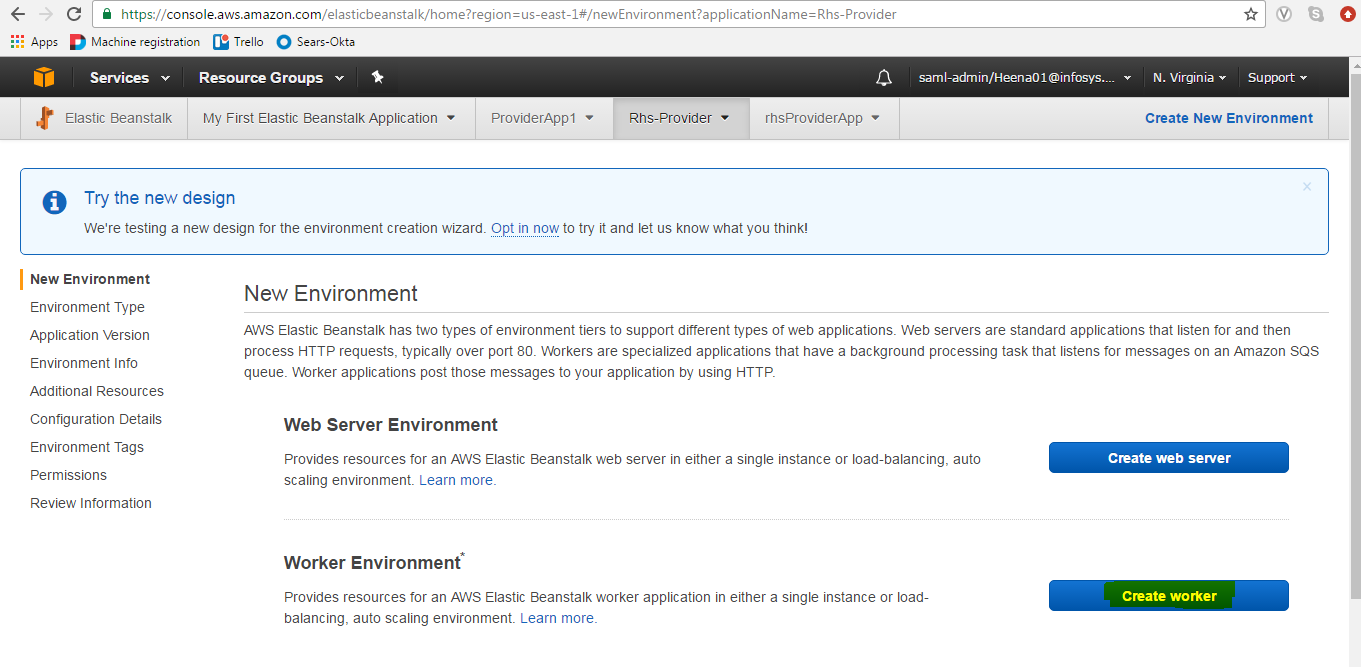
Create a new application in elastic Beanstalk and follow the steps as described in the below screenshots.

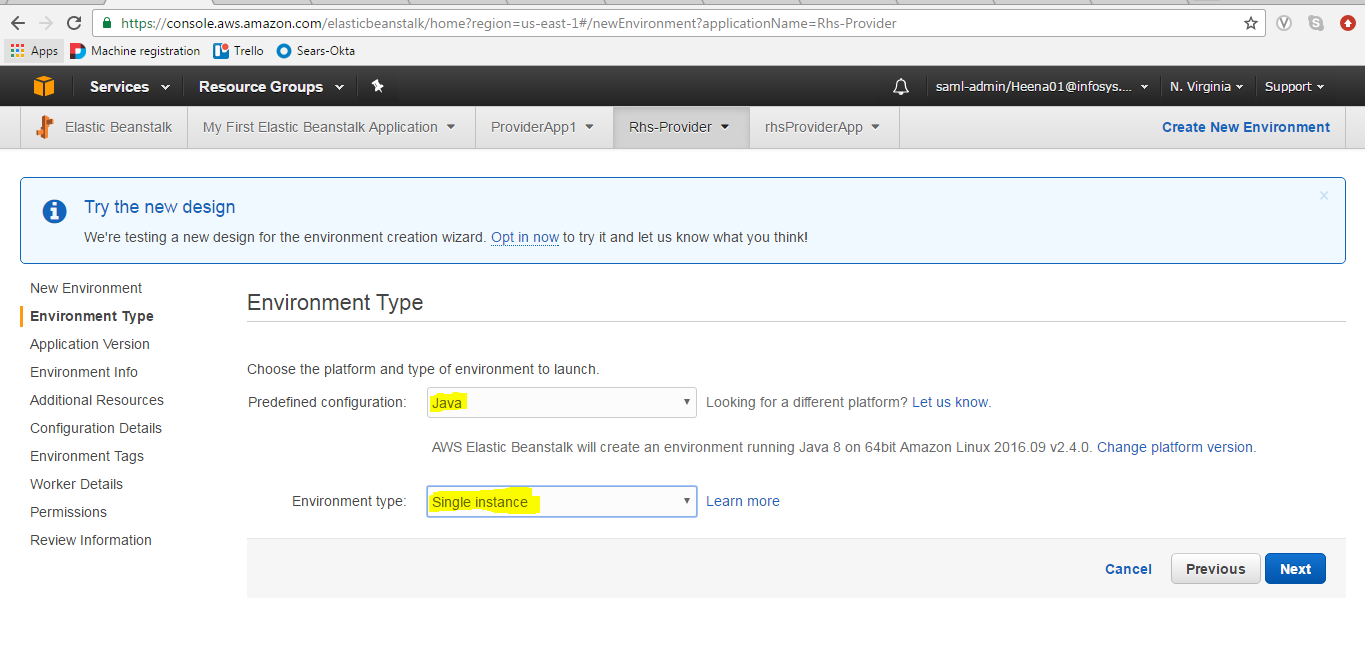
Here we will choose the upload method (from local system). We can also specify S3 location from where the beanstalk will directly pick the deployable archive.

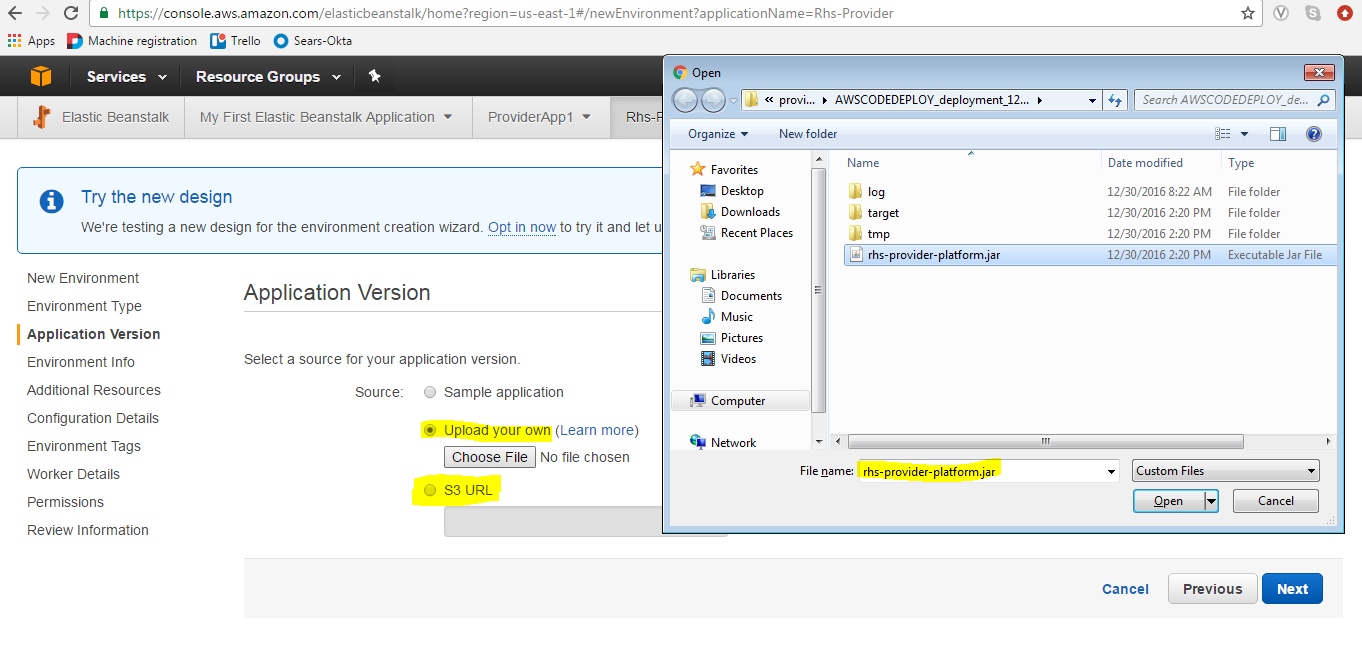
Also while creating application, if we do not specify the role and security group, beanstalk will create default ones and do the deployment. For viewing the application in browser we will have to modify the security group settings and add a custom TCP rule to allow outside world hit the machine’s 9090 port through TCP. That setting has been done for the security group. Rest, all settings are default.

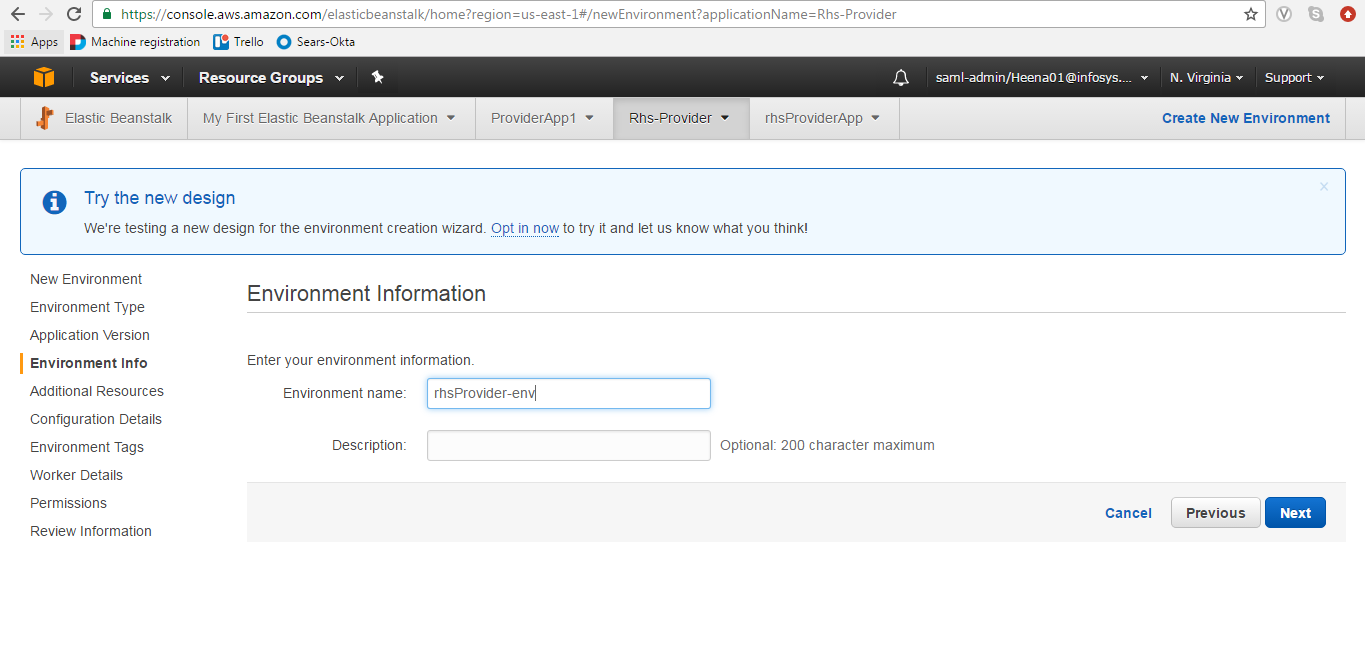


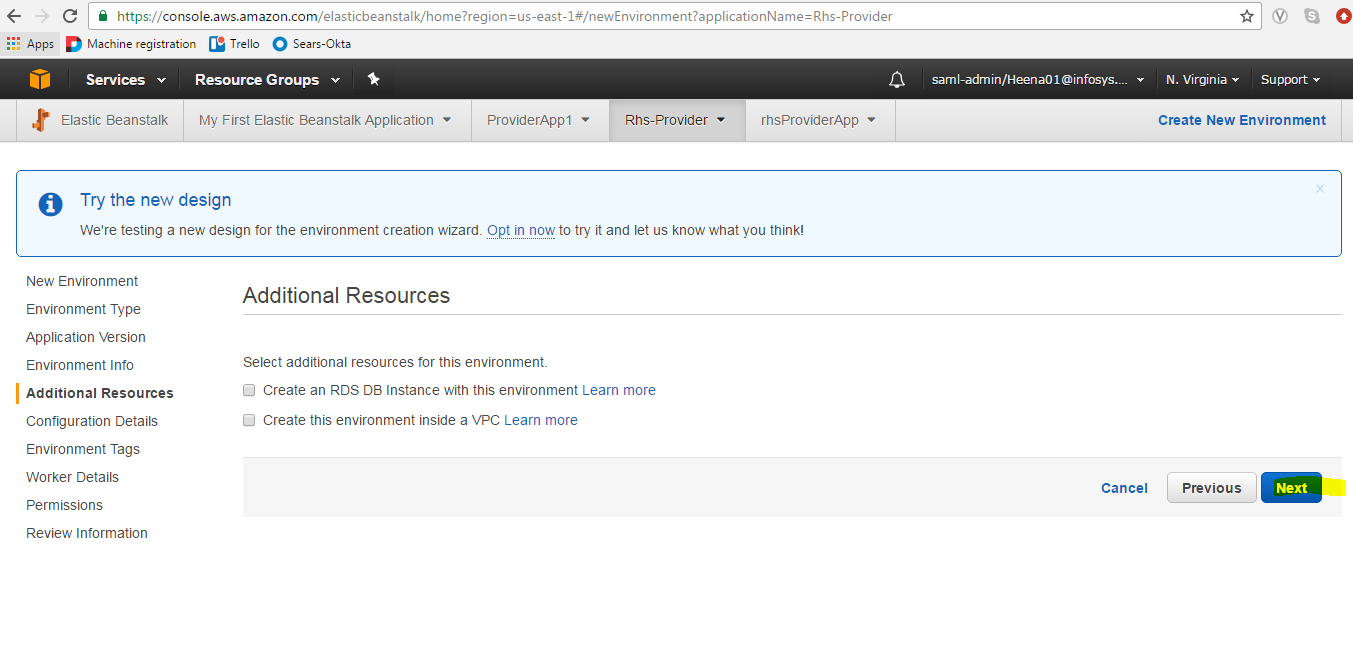


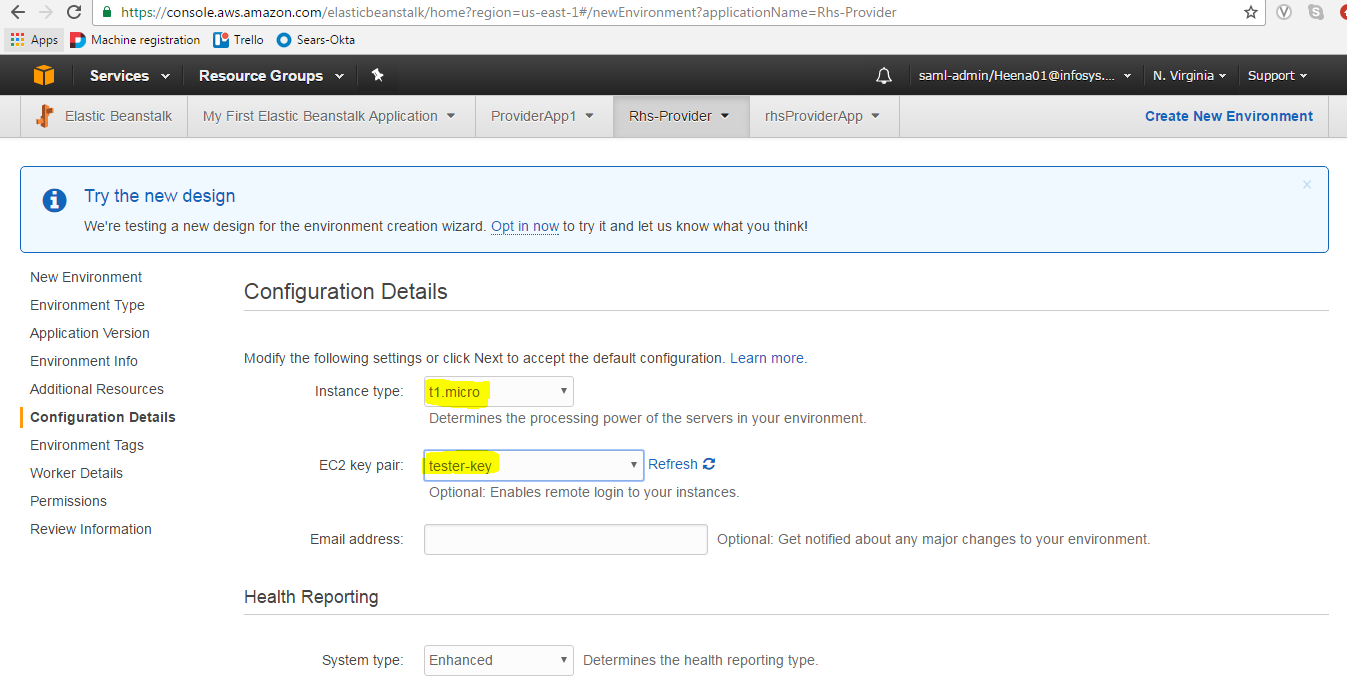


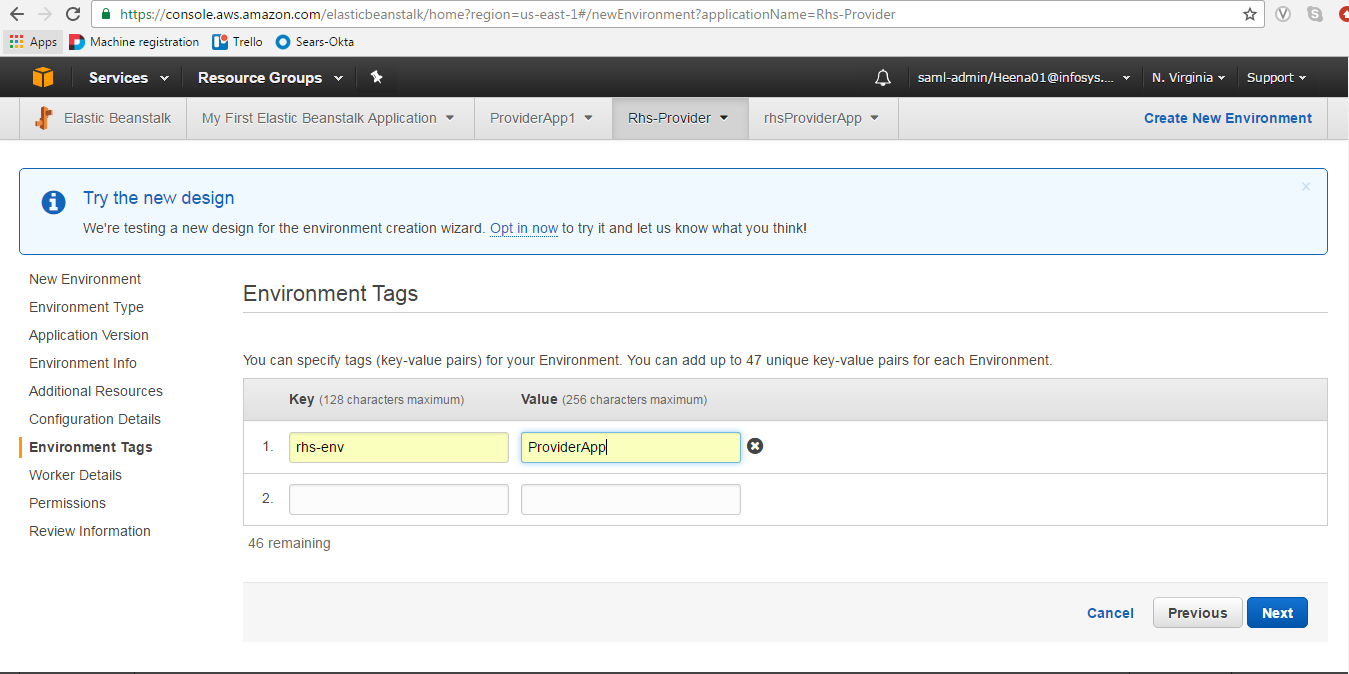


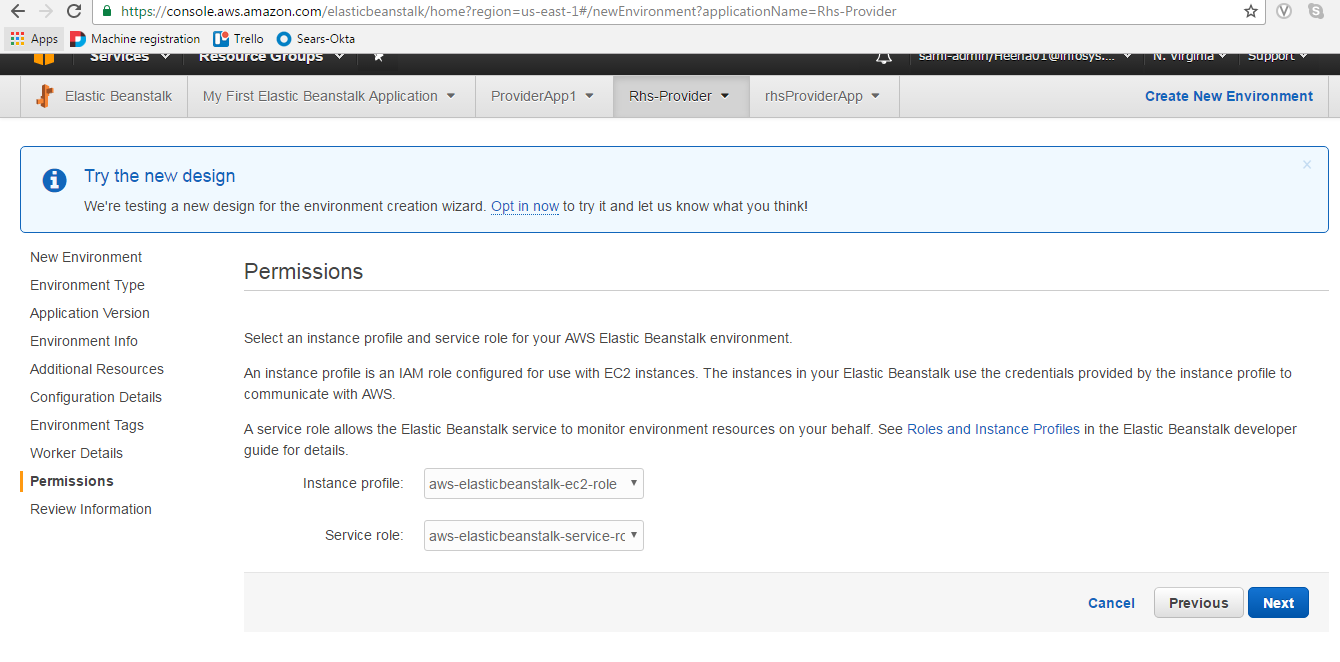






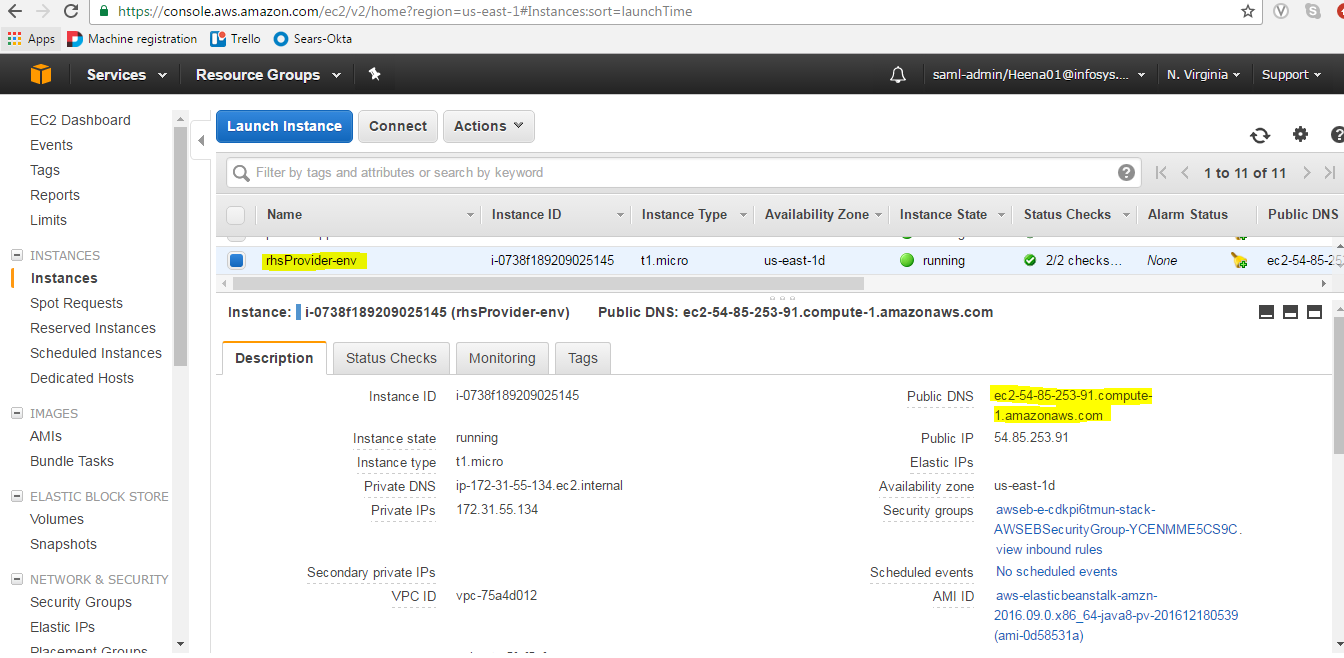






Lastly Review and Launch.

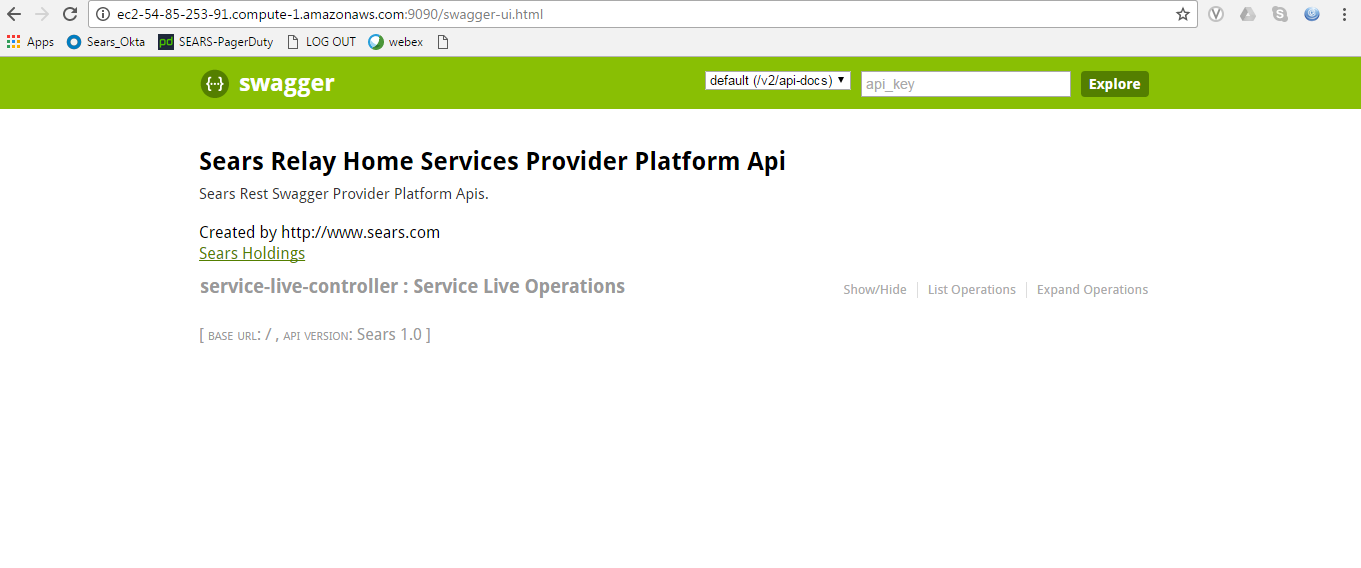
Once the application is launched and is up, as from the Elastic Beanstalk dashboard, to verify the same, go to ec2 dashboard and choose the ec2 instance created by the beanstalk application.



Check the details and copy the public DNS name of the instance and in the browser hit the url after appending context root in the url after the port number, eg here :9090/application/welcome.

So the complete url becomes :

<http://ec2-54-85-253-91.compute-1.amazonaws.com:9090/application/welcome>



References : <https://aws.amazon.com/elasticbeanstalk/>