



Tix Technologies Migrates to AWS Cloud with Rapyder

Introduction

TIX was founded in 2009 with the single goal of making live events more affordable in Australia. They achieved this by launching Lasttix – a destination for discount ticket offers to theatres, concerts, musicals, and more.

Fast forward 10 years, and they have expanded their goal also to make live entertainment more accessible and have done this by launching the TIX app, which, in one straightforward interface, helps its users instantly identify the best ticket prices no matter where they are or what their budget is.

TIX now operates in 4 countries with over 1.6 million members

Business Need

CUSTOMER had developed a .NET application for a ticket-selling product, built using Windows Stack, and wanted to host this application on the Cloud. After discussions with the Rapyder team of Cloud engineers, they decided to go with AWS to benefit from the flexibility of Infrastructure sizing and to have a Scalable infrastructure along with High Availability for their different components, as they were expecting massive traffic on the platform.

They also wanted to have CI/CD pipelines for each of their application components.

» Third Party Solutions

Microsoft SQL Server
Enterprise 14.0.3023.8

» AWS Services

EC2, ALB, S3, CloudWatch, CloudTrail, GuardDuty, Config, EC2, Auto Scaling, Load Balancer, S3, AWS AD, Guard Duty, Code Pipeline, Code Deploy, Lambda, SNS, Work Mail, Systems ManagerCloud Endure, Systems Manager

Solutio Approach

To address the business objectives that needed to be achieved, Rapyder implemented a solution on AWS that, considering the future growth expected, provided Tix Technologies with a flexible and scalable environment:

- » The application has UI, API, and DB layers. All the servers were launched in a private subnet to prevent the servers from being exposed to the public directly.
- » High availability was factored in, and servers were launched in HA mode across different availability zones.
- » UI and API layers were deployed in Auto-scaling groups behind the Application Load Balancer to meet the scaling demand of the Customer.
- » MSSQL enterprise edition was used as DB to run the database in Always-On mode so that high availability would be taken care of.
- » AWS-managed Microsoft AD was implemented, and all the servers made part of the AD to manage authentication centrally.
- » Servers using IAM roles to access the services like S3 and Cloudwatch.
- » IAM users created and only required access provided by using IAM policies.
- » Custom metrics were set up on all servers, and monitoring was enabled by using Cloudwatch.
Cloudwatch alarms were also created for CPU, Memory, and Disk Utilization.
- » Antivirus was installed on each server to protect the servers from malware attacks.
- » AWS CloudFormation is used for spinning up the entire infrastructure.
- » CI/CD pipelines have been set up for each application component. AWS CodePipeline and CodeDeploy were used for CI and CD.
- » Jenkins hosted on EC2 machine.
- » AWS Guard Duty was enabled, and a custom solution was developed using lambda scripts to take action on the findings.

- » Custom Lambda function was also written to notify the Customer of CI/CD.
- » CloudTrail was enabled for all regions.

Reaping Rewards

- » Tix Technologies is now successfully running its workloads on AWS Cloud, which can handle the massive traffic with the help of Auto-Scaling groups.
- » All their deployments are automated, and there is no human intervention the same which has reduced their go-to-market.
- » The servers are near auto-healable as the environment is configured to be running in high availability (behind the load balancer and in Auto Scaling groups). Thus, there is no downtime for the applications.

