Case Study: DevOps Implementation

Cync Software is a financial solutions company based in Tampa, Florida; they deliver complete software solutions for commercial finance companies as well as banks that provide asset-based loans. Their flagship product, Cync Applications Suite, offers a diverse collection of financial solutions that cover a vast range of account receivable financing, factoring, working capital loans, asset-based lending, and related credit services.

CHALLENGE:

As a key player in the commercial lending market, Cync software's primary challenge was to implement an automated delivery mechanism for their diverse software solutions release cycle that can rapidly deploy, and scale based on the trending needs of their customer base. While they increased the pace at which they developed new capabilities, the team was not able to push their code to production at an accelerated pace. As a matured cloud native application, Cync's Financial Lending software platform needed to have a "Faster", "Secure" and "High Quality" release cycles. As they continue to expand and add more customers to their portfolios, increased requirements meant multiple and frequent releases with continuously changing technology platforms.

The leadership team hired Idexcel to implement a DevOps pipeline so that the software release and development cycles were better aligned. Cync wanted to be sure that they were better positioned to react to customer requirements.

Some of the technical challenges included:

- **Cloud Native** – Since the application was cloud native, the pipeline had to be secure, self-healing and cloud native.
- **Automation** – The deployment and validation needed be automated including provisioning of infrastructure (immutable), Code Testing and Code Deployment.
- **Containerization** – Application was using Docker, therefore pipeline had to be capable of triggering deployments within Containers.
- **Security** – Since the application was in the financial domain, Security was the top priority. Automation for Anti Viruses to Static/Dynamic Code Analysis and SOC2 Compliant policy checks needed to be implemented.
- **Tools** – All the tools had to be compatible with the AWS Platform and fully integrated with each other.
- **Logging** – All the events needed to be logged. Real-Time and Archives had to be maintained for pre-determined time period.

SOLUTION:

Idexcel assigned a team with development & delivery expertise on the AWS cloud to lead the project; they understood the finer details of both application development and system administration. The team designed a pipeline that consisted of both AWS Services and Open-Source Software.

The Implemented CI/CD Pipeline included:

- **Cloud Native** – Pipeline used AWS Native services like - CodeDeploy, S3, SNS, CloudWatch, Lambda and CodePipeline
- **Automation** – Automated Provisioning of infrastructure was achieved using Ansible scripts, Code Testing using iRAF" (Idexcel's QA Automation Framework) and Code Deployments achieved using Opsworks + Chef Automate.
- **Security** – Two-fold approach was implemented using automated scripts. The scripts scanned, Infrastructure and Code for vulnerabilities and presented a report prior to deployment.
- **Tools** – The tools identified included – Static/Dynamic Code Analysis **Brakemen** (RoR), and **AWS Inspector** (AngularJS). Containers – **Docker**. Infrastructure Scanning - **MBSA** (Windows), **Fortigate** (Linux). Configuration Management and Deployment – Chef. Finally, QA Automation **Selenium (Cucumber) + Jenkins.**
- **Logging** – "Logs & Live Monitoring" approach was adopted. AWS Native Services – **CloudWatch** and **Kinesis** were used with customized **SNS** notifications.