International Journal of Engineering Applied Sciences and Technology, 2020
Vol. 5, Issue 2, ISSN No. 2455-2143, Pages 156-158
Published Online June 2020 in IJEAST (http://www.ijeast.com)

DevSecOps: A CASE STUDY ON A SAMPLE IMPLEMENTATION OF DevSecOps

Avanish Parashar
Department of CSE
IMS Engineering College, Ghaziabad, U.P., India

Aditya Dwivedi
Department of CSE
IMS Engineering College, Ghaziabad, U.P., India

Anand Kumar
Department of CSE
IMS Engineering College, Ghaziabad, U.P., India

Atif Ahmad Khan
Department of CSE
IMS Engineering College, Ghaziabad, U.P., India

***Abstract*— It is a critical task to integrate security into DevOps as conventional security methods have failed to keep up with fast development operations. In DevSecOps we address this problem by integrating security with the development operations through various phases of the development life cycle. The purpose of this paper is to give the basic idea about DevSecOps, its benefits, implementation and challenges during the process of implementation. So, we did a sample implementation of DevSecOps & learned its utility in contrast with conventional security methods & the challenges faced while its implementation.**

Keywords— **DevSecOps, Jenkins, Docker, CI CD pipeline**

I. INTRODUCTION

"DevOps" is an abbreviated form of the words "development' and "operations ".[i] DevOps explains the procedures to pace up the processes from development to deployment in the production cycle.
In the collective model of DevOps, security is not a separate phase; it is instead a shared responsibility. This led some to invent the term "DevSecOps" to emphasize the need to build a security foundation into DevOps initiatives.[ii]

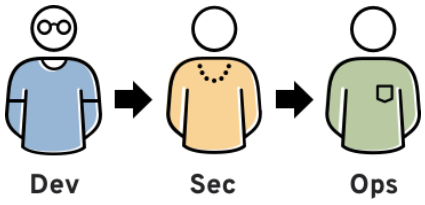


Fig 1: by redhat.com

II. PRINCIPLES OF DEVSECOPS

One of the basic purposes of DevSecOps is to automate security operations. It aims to create a culture of "Security as Code" thus promoting synergy between developers and security team. It deals with the problem of collaborating security with continuous delivery pipeline and share the responsibility of security tasks across the development lifecycle.[iii]

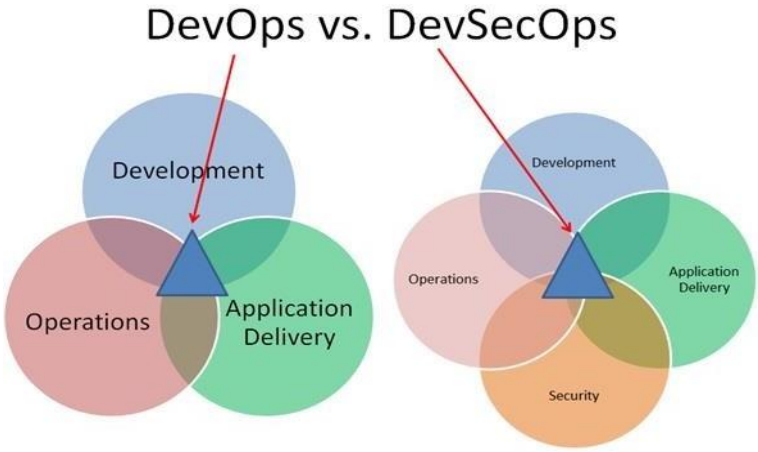


Fig. 2: by developer.ibm.com

The DevSecOps approach has 5 major principles[iv] to ensure an effective process.

A. Adopting an empowering attitude

B. Building up security

C. Including continuous learning

D. Sharing threat intelligence

E. Promoting open collaboration

International Journal of Engineering Applied Sciences and Technology, 2020
Vol. 5, Issue 2, ISSN No. 2455-2143, Pages 156-158
Published Online June 2020 in IJEAST (http://www.ijeast.com)

III. DEVSECOPS TOOLS

A. *Jenkins*

Jenkins provides a simple and easy way to continuous delivery (CD) and continuous integration (CI) environment for almost any blend of languages.[v]
Jenkins attains Continuous Integration with the aid of plug-ins. It permits the integration of various stages of DevOps.[vi]

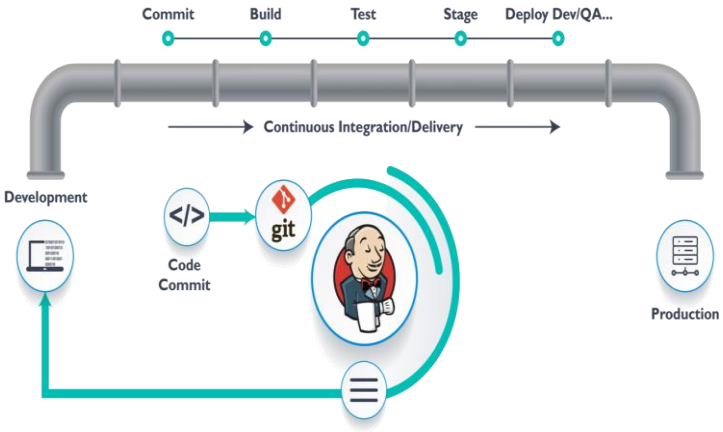


Fig 3: by dzone.com

B. *Apache Tomcat Server*

Apache Tomcat was developed by the Apache Software Foundation (ASF). It is an open source web server and container. It is used to implement various Java EE functions including JavaServer Pages (JSP), Java Servlet, and WebSocket, and provides a java server environment wherein Java code is executed.[vii]

C. *SonarQube*

It is a tool which makes it simple to integrate the findings or statistics into the build pipeline. It is used to continuously inspect the code quality and guide the developers while code reviews.[viii]

D. *OWASP Dependency Checker*

It is a tool which recognizes the dependencies of the project and finds out if there are any vulnerabilities of any type. This plug-in can independently execute a Dependency Check study and observe results.[ix]

E. *Docker*

It is a utility used to build, deploy and run the project via containers. It is a virtual machine, but not like virtual machines that build a totally distinct operating system. It allows the programs to use the Linux kernel of the very machine on which it is installed and by taking this advantage, it can cause the programs ready to ship to other machines that are running the same Linux OS with different configurations. Hence, program size is reduced remarkably and performance also gets improved at the time of shipping.[x]

IV. CHALLENGES IN IMPLEMENTATION

A. *Integrating security into the infrastructure*

- Ports and protocol hardening through Terraform
- Linux CIS hardening through Puppet
- Baseline hardening through Dev-Sec Hardening Framework

B. *Integrating security into the application*

- Major companies spend significant time protecting applications from threats.

C. *Integrating security into the CI/CD pipeline*

Regarding integrating security in the production process, there is a lot to implement in CI/CD pipelines.

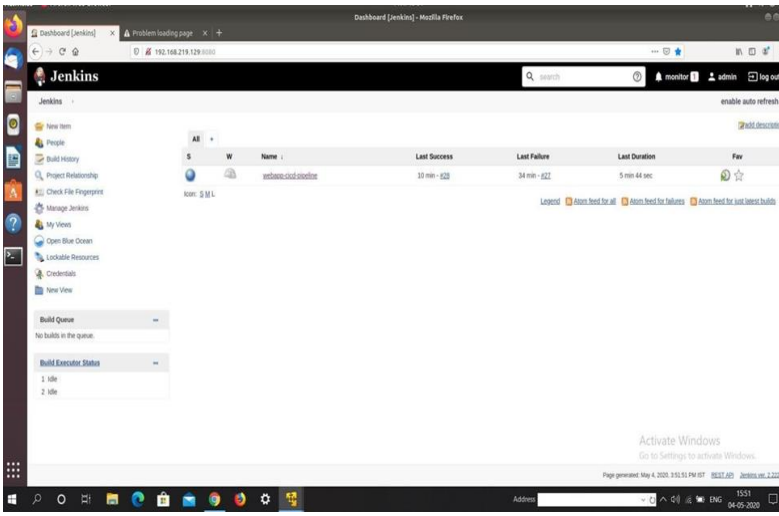
- Container testing
- Security hardening
- OS network hardening

D. *A common problem: Protecting the border*

- The problems with cloud-native appliances
- The problems with traditional appliances
- Getting cloud-native security that supports DevSecOps

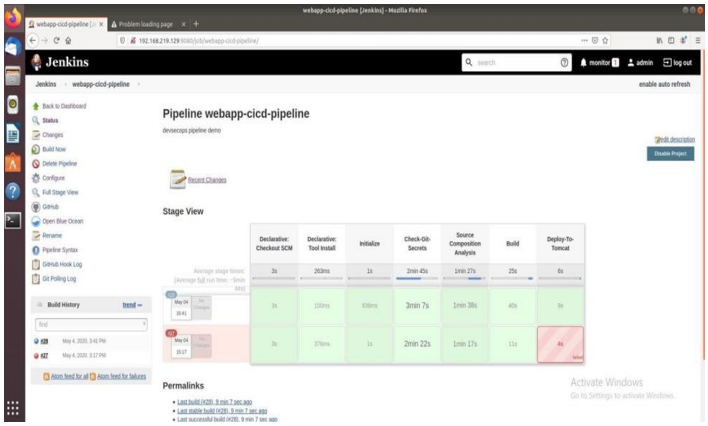
V. IMPLEMENTATION

A. *DASHBOARD OF JENKINS*

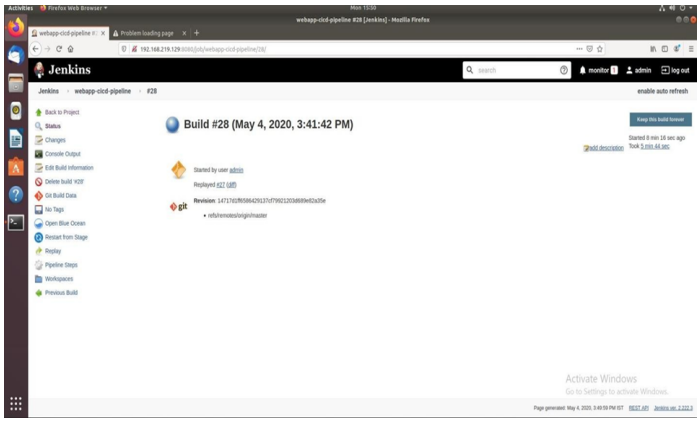


International Journal of Engineering Applied Sciences and Technology, 2020
Vol. 5, Issue 2, ISSN No. 2455-2143, Pages 156-158
Published Online June 2020 in IJEAST (http://www.ijeast.com)

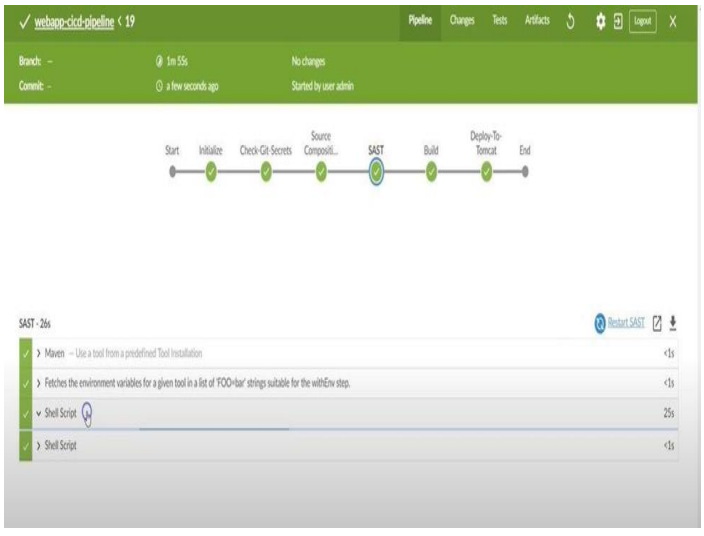
B. CI/CD PIPELINE BUILD



C. BUILD COMPLETED SUCCESSFULLY



D. VIEW OF VARIOUS STAGES OF PIPELINE USING BLUE OCEANIC PLUG



VI. CONCLUSION

This presents the research we did on DevSecOps to find out how DevSecOps can be implemented, what doing DevSecOps means for an organization with regard to what principles and practice they should adhere to, what challenges they would face attempting to adopt DevSecOps, the benefits if it's done successfully and how it has evolved from the need to implement security in DevOps to what could seem like a movement on its own.

We found that DevSecOps is defined by many as the integration of security processes and practices into DevOps environments, that DevSecOps promotes a set of principles meant to shift the mindsets of all participants in the software development process so everyone participates and do what they can to ensure security in the project and a set of practices that can ensure security in the project based on the idea of planning and implementing security from the start and as code.

We identified a set of challenges and benefits to implementing DevSecOps. The challenges we identified should not be seen as deterrents to implementing DevSecOps, but a symptom of its youth. As DevSecOps matures, better methods, practices, tools etc. can probably overcome them. The benefits we identified indicates it is maturing, by for example resulting in less unplanned work and a decrease in manual labor.

VII. REFERENCES

[i] 'What is DevSecOps ?' redhat.com https://www.redhat.com/en/topics/devops/what-is-devsecops

[ii] [ii] 'What is DevSecOps ?' redhat.com https://www.redhat.com/en/topics/devops/what-is-devsecops

[iii] Gilad Maayan, August, 2019 ibm.com https://developer.ibm.com/recipes/tutorials/devsecops-security-and-devops-working-together/

[iv] Gilad Maayan, August, 2019 ibm,com https://developer.ibm.com/recipes/tutorials/devsecops-security-and-devops-working-together/

[v] Martin Heller, March, 2020 https://www.infoworld.com/article/3239666/what-is-jenkins-the-ci-server-explained.html

[vi] Saurabh, May, 2019 https://www.edureka.co/blog/what-is-jenkins/

[vii] 'Tomcat' xebialabs,com https://xebialabs.com/technology/tomcat/

[viii] 'SonarQube | SonarSource' https://www.sonarsource.com/products/sonarqube/

[ix] 'OWASP Dependency Check' https://plugins.jenkins.io/dependency-check-jenkins-plugin/

[x] Janbask Training, February, 2020 https://www.janbasktraining.com/blog/what-is-docker/