DevSecOps

Producing high quality, secure software at pace

Enterprises across the world are demanding software release at high speed to meet business requirements. when software is developed at such speed, security should not be left behind which can only happen if security is built in to SDLC. Such requirements gave birth to technologies like Agile development, DevOps and DevSecOps.In this paper we describe the DevOps technology that enables Development team and Operation team to collaborate with each other on day to day basis such that operational issues and customer problems reduce to a larger extent. We also explain DevSecOp technology that allows security to be built in to the application through automation, cultural shift, application security programs, etc. In the end we describe what technologies and tools NIC is providing to the developers to implement DevSecOp across organisation.

We must always meet customer's requirement. Be it any role in the software industry - developer, tester, security auditor or manager, our job is to support the business so that it wins in the marketplace. Now, there is tough competition among the business players in every field to woo the customers. So, they demand product innovation and delivery at a rapid pace. Three or four product releases in a year is no longer a norm, business demands the release every week or every month with new features or to support the customer requirements. These paradigm shifts happening in the industry gave birth to technologies like Agile Software Development practices, DevOps, DevSecOp, etc.

Why DevOps?

If Development and Operation work in silos, then when a developer writes code, builds it, tests it and deploys it into the operation, it normally fails. Whatever the failure may be- deployment failure, operation failure or crashes, the customer faces the problem in running the business. Normally, in such cases, the blame game begins, people from development say that there are operational issues and people from operations blame it on development issues and a lot of time is lost in the process. This usually happens because development and operation are not in sync with the software stack, tools and versions. Moreover, as discussed earlier, it is today's requirement to push the code to deployment/operation at a rapid pace, certain deployment each day. DevOps was created to address all these issues.

DevOps best practices can be narrowed down to three basic principles called the three ways :

Anil Kumar Jha
Sr. Technical Director
(Application Security Group)
aniljha@nic.in

First way

The first principle says we need to accelerate the work from development to operation, and then to the customer. This can be achieved by limiting work in progress and through automation.

Second way

The second way enables constant flow of feedback from operation to development. This can be achieved through continuous integration, build and deployment process working together with a fast, automated suite of tests.

Third way

The third way is about creating a culture of continual experimentation and learning.

While DevOps culture brought a lot of innovation to the software development process, security was either not considered or not able to keep pace with the rate at which software was being built and released. DevSecOp is an attempt to inject security into the DevOps process and make sure that software delivery rate is not disturbed due to this injection.

DevSecOp is short for Development, Security and Operation. This technology dictates that security is not the job of one group, rather everyone including development, security, operation quality is responsible for security. If any issue comes in production, all the teams should work together to resolve the issue therefore all the teams should learn security. Security should not be an afterthought, rather it should be built into the application. Security should be discussed and practiced during all the phases of Software Development life cycle during requirement analysis, architecture, design, development, testing and in operation.

Hence, to deliver software at high speed and make it secure as well, security needs to be built into the application development workflow and process. The earlier we introduce security into SDLC, the sooner we will be able to identify and fix vulnerabilities in the software, rather than waiting till the end for security assessment reports or run time issues in the operation. Organizations can introduce security into existing continuous integration and continuous delivery (CI/CD) pipelines. Just like after a build failure, software is not eligible for deployments in the production, a policy may be defined by the organization, if a security issue is caught in the CI/CD pipeline, the application should not reach production until the vulnerabilities are taken care of.

To implement DevSecOp, organizations need to integrate application security tools into CI/CD pipeline.

Some of the tools are:

- Static Application Security Testing (SAST)
- Software Composition Analysis (SCA)
- Dynamic Application Security Testing (DAST)

SAST

SAST tools scan the source code of the application and identify security vulnerabilities that may be exploited by the hacker while in production. These tools pinpoint the code location where security issues exist. This tool is used by the developer while the developer is writing the application and so the advantage of this tool is that the developer may fix all the security issues reported by the tool during the development phase of SDLC.

SCA

SCA tools scan the source code and binaries to identify vulnerabilities in open source libraries included in the application. These tools not only uncover the security issues but also highlights the licensing risks due to open source software components.

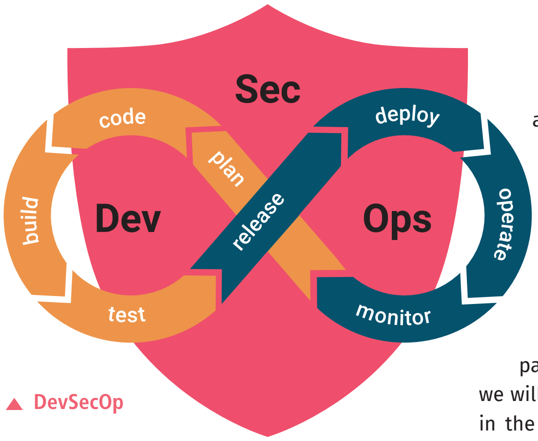
DAST

DAST is an automated black box testing technology that attacks our web applications/APIs just like hackers would do. DAST tools do not require access to source code to perform the scan on web applications/APIs. Since these tools attack the application in real time just like a hacker and provide proof for each of the reported issues, it has a low number of false positives.

> With traditional software development strategies, it is not possible to deliver secure software at high velocity. DevSecOps is the philosophy of incorporating security practices within the DevOps using automation and security tools, and thus enabling secure software delivery through the seamless and transparent integration of security into the CI/CD pipeline.

RATNABOLI GHORAI DINDA
Dy. Director General, NIC

In NIC, the Application Security Group is providing the following tools to developers to enforce security in the CI/CD pipeline through automation:



DevSecOp

HCL Appscan Source for Development

This tool provides a plugin to integrate in Developer IDE like Eclipse, Microsoft Visual Studio, etc. This allows developers to scan the code, find vulnerabilities and fix it during the development phase of SDLC. This tool also provides recommendations for the issues reported by it.

HCL Appscan Enterprise

This tool is a black box security testing tool that identifies the security threats by attacking the running web applications/APIs/web services just like hackers. It does not require source code of the application and can scan the applications developed in any language. It also provides proof of the reported issues of what parameter tampering/manipulations/fuzzing were done and its effect on the target application.

Conclusion

If we utilize these tools regularly as a part of our design and development work, we will be able to eradicate security issues early in the SDLC. Moreover, the developers acquire a lot of security knowledge in the process and will apply these learnings in the other projects they work up on. All these things will certainly make a cultural shift in the organization and will ultimately make the software much more secure, robust and resilient.

For further information, please contact:
Anil Kumar Jha
Sr. Technical Director
Application Security Audits & Assessment Division
National Informatics Centre, A-Block, CGO Complex
Lodhi Road, New Delhi - 110003
Email: aniljha@nic.in, Phone: 011-24305140