# Injecting security in CI CD pipeline

DevOps is pretty much known for the path-breaking changes it has brought in the software industry. The most striking has been to bring the Dev and Ops team to work in sync at all times throughout the application development lifecycle. And the second is to automate pretty much the entire CI/CD pipeline. These two have been the most remarkable transformation brought to us by the DevOps process.

DevOps has been constantly finding ways to make the CI/CD pipeline more efficient and better. Today we are going to discuss DevSecOps; the process aims to put a security blanket around the entire lifecycle.

In the older processes, the security check-in the older processes for the application used to happen at the later stages usually before the deployment. This practice would result in fixing last-minute code and testing issues which in turn delays the product release.

 So, to accelerate the whole process DevOps uses the ‘Shift Left’ approach. The method focuses on bringing the security practice right from the early stages of the DevOps lifecycle. The key is to incorporate test and security at the beginning ensures speedy process.

Securing the application is not which means to be done at a certain point it needs to be done at every step throughout the process. Securing the application is a continuous process that’s why it’s called Continuous Assurance.

## Now the question arises what do we need to check for the security?

There are several security checks need to be performed for

**Source Code Vulnerabilities**- This check is related to security of the software. If the source code is not protected might be subjected to potential malicious attacks.

**OSS Library Vulnerabilities**- Well not just source there are high chances that the open source library used in the application can have vulnerabilities.

**OSS Version**- Open-source libraries come in handy, but there are chances that after a few years that version may be deprecated. If deprecated then there might not be any maintenance or any replacement for the library.

There are several other vulnerabilities that the application might be exposed to due to libraries, code infrastructure, or any exposures. So here are some methods to how do we check for security?

**Static Application system Testing (SAST**)- The testing is primarily done before code compilation. The testing method analyses the code security vulnerabilities. It is also known as white-box testing. This test happens very early in the SDLC as it helps to fix the code issues.

**Active and Passive penetration test (Dynamic Analysis) -** The test is described as a dynamic analysis because it checks the system response to variables/parameters that are not constant. In easy language, it checks the application behavior with real-time values.

**Infrastructure Analysis-** This involves scanning the actual environment like configuration, server status to understand and analyze the actual drift and what could be the fix for the drift.

These are some of the checks that are performed by the team to ensure a secure CI CD channel. There are many tools available in the market to perform these tests. Let’s have a look at a few tools:

**Checkmarx**- Facilitates the SAST testing to analyze the code vulnerabilities in the early stages. It can be easily integrated with any CI/CD tool or environment.

**IMMUNIO-** The tool provides cloud based solution to protect the web application from malicious attacks. The tools is unique because it does not continuously scan the application instead it focuses on possible vulnerabilities.

**Aqua Security-** The tool gives the security for containers throughout the CI/CD pipeline. The main feature is that it works with all platforms and clouds very well.

**OWASP Zed Attack Proxy (ZAP) -** One of the most popular tools to protect the web applications from potential threats. It produces ZAP Docker weekly which has all the common vulnerabilities listed.

**Twistlock** - A multifaceted tool which offers security to containers, hosts, and serverless components.

The list goes on as there are many more tools available as per the need of the application.