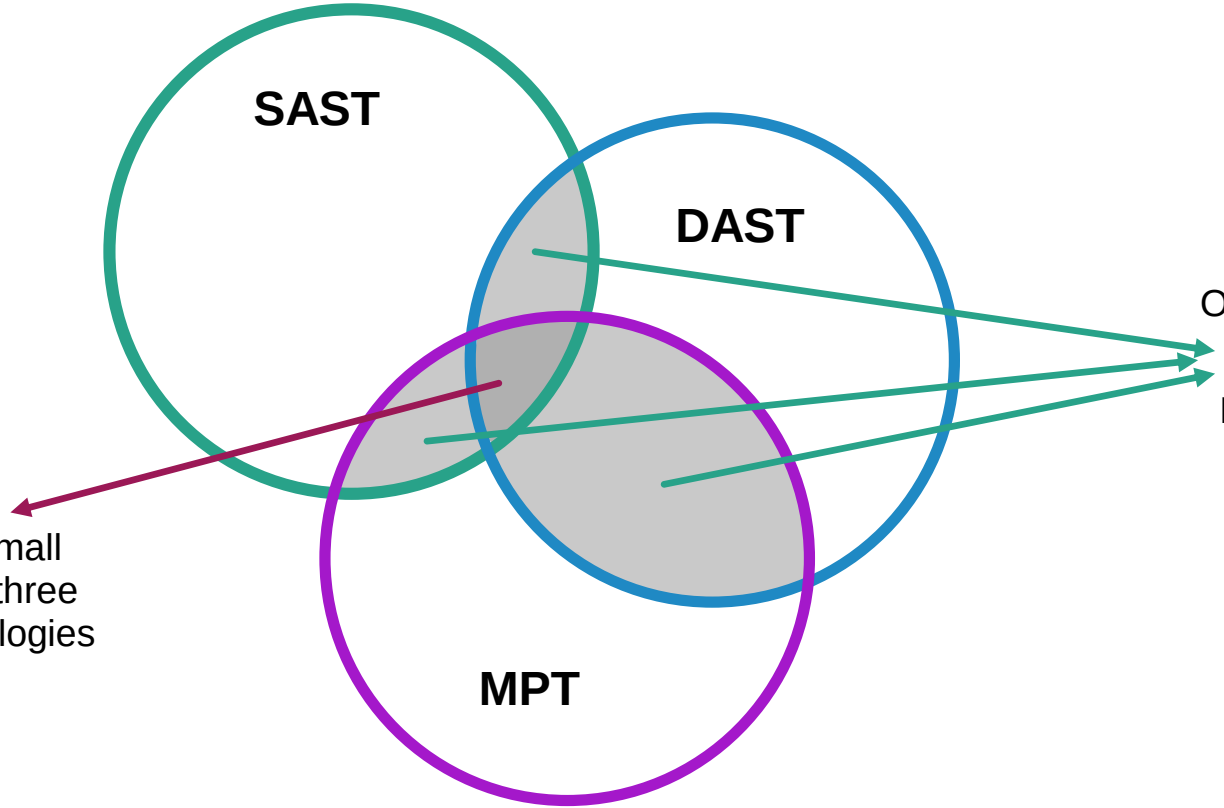

IT-SA 2019

# Making Applications and Docker Containers Secure

@d3v_rand0m
Julian Totzek-Hallhuber

# What is application security testing?




Overlap of the different technologies.

**Each technology is able to identify different things**

Only a very small overlap for all three different technologies

# HOLISTIC APPROACH TO APPLICATION SECURITY

**BINARY STATIC ANALYSIS (SAST)**

Conduct "MRI for Vulnerabilities" Without Needing Source Code

**Greenlight (SAST)**

Security Unit Testing

**SOFTWARE COMPOSITION ANALYSIS**

"Bill of Materials" to Identify Open Source & 3rd Party Components with Published Vulnerabilities

**VENDOR APPLICATION SECURITY TESTING (VAST)**

Eliminate Weak Links From 3rd Party Commercial Software

**DYNAMIC ANALYSIS (DAST)**

Probe for Vulnerabilities by Emulating Cyber Attacker Techniques

**WEB APPLICATION PERIMETER DISCOVERY**

Rapidly Discover & Assess Risk for All External-Facing Web Apps

# 1st party code vs 3rd party code

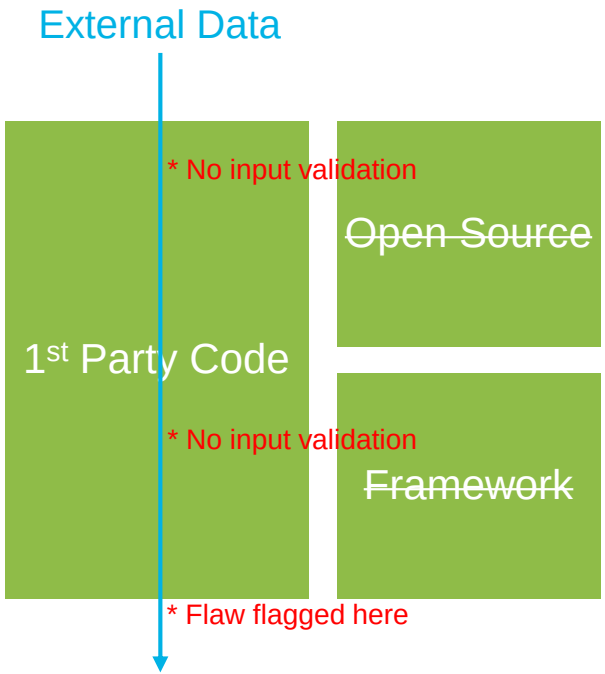
**1st party code**

- Is what you write
- Is what you know about your application
- Is what you need to fix
- Makes your life hard

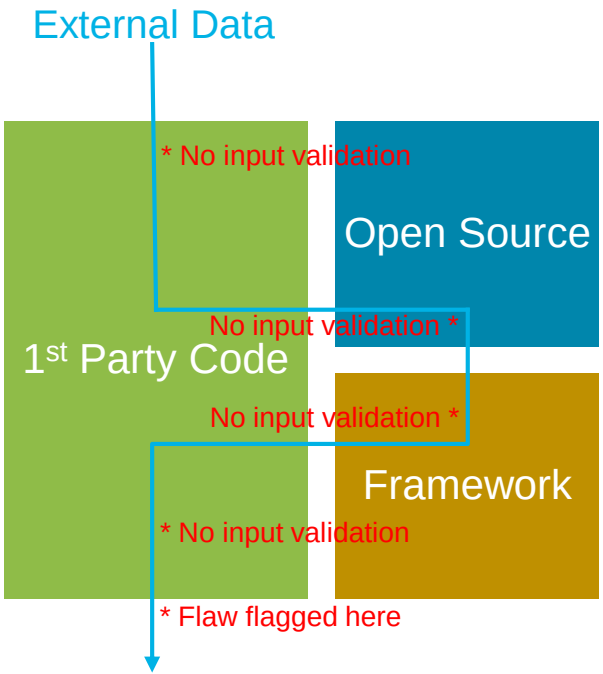
**3rd party code**

- Is what others write
- Is what the world knows about your application
- Is what you will never fix
- Makes your life easy
- Refered to as libraries or components
- Open source in many cases

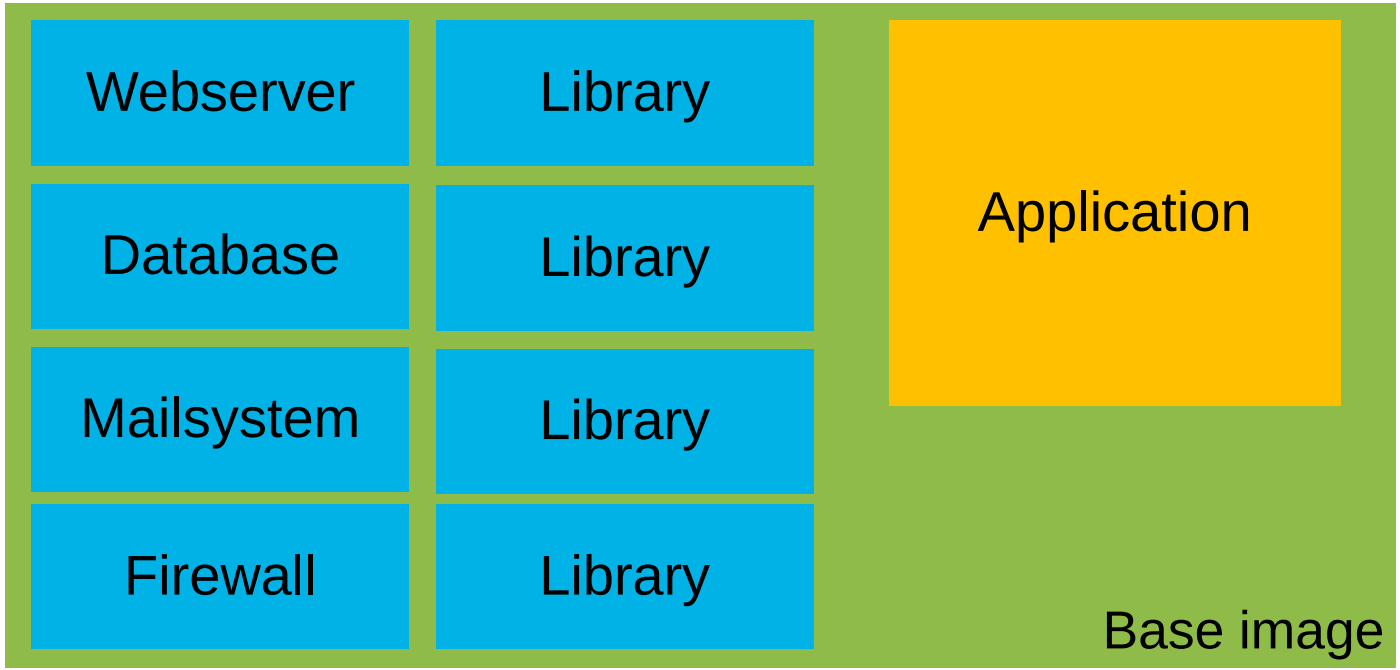
# Source Code Analysis vs Binary Static Analysis



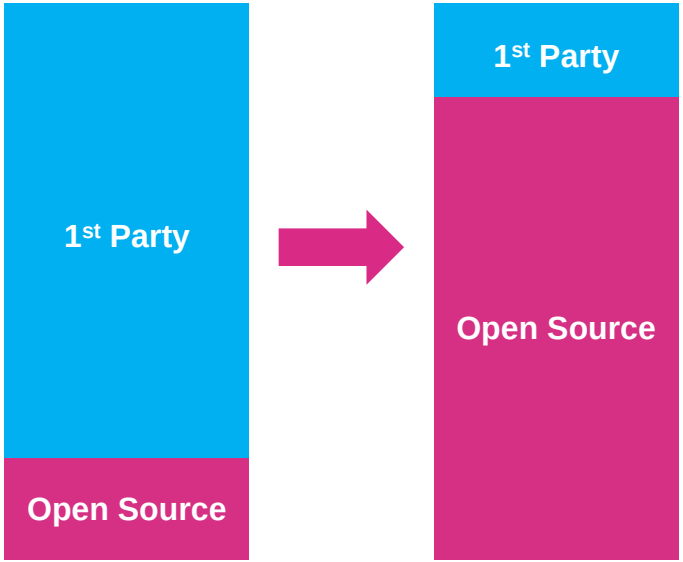


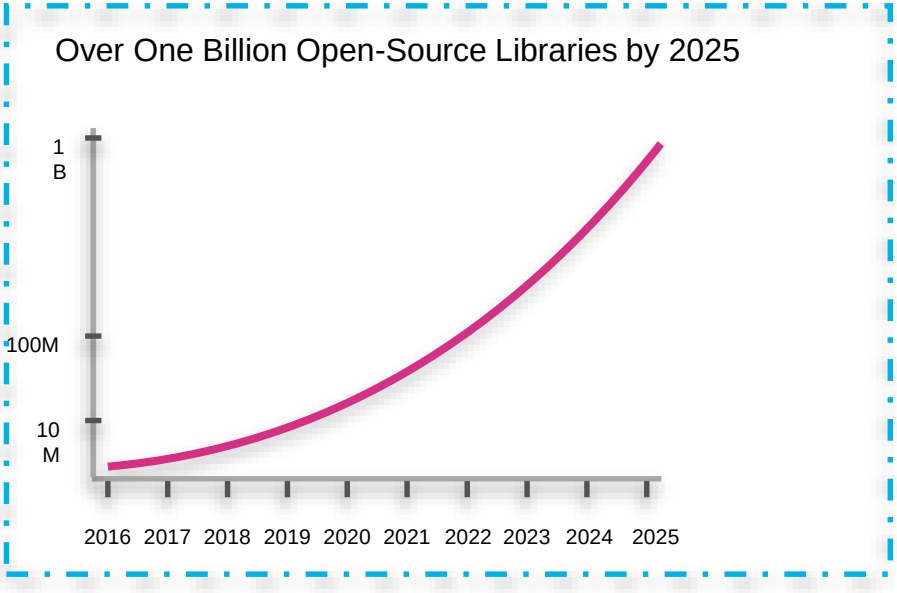


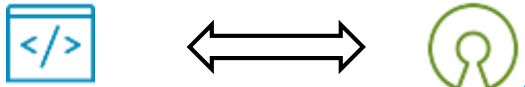
# Scan Docker Containers?



# Software is Increasingly Assembled



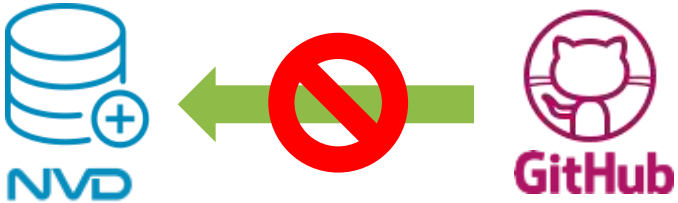



# Open Source Is Breaking the NVD Model

Silent Fixes

- NVD was designed for a different era
  - Fewer large commercial vendors
  - Manual, tightly controlled process



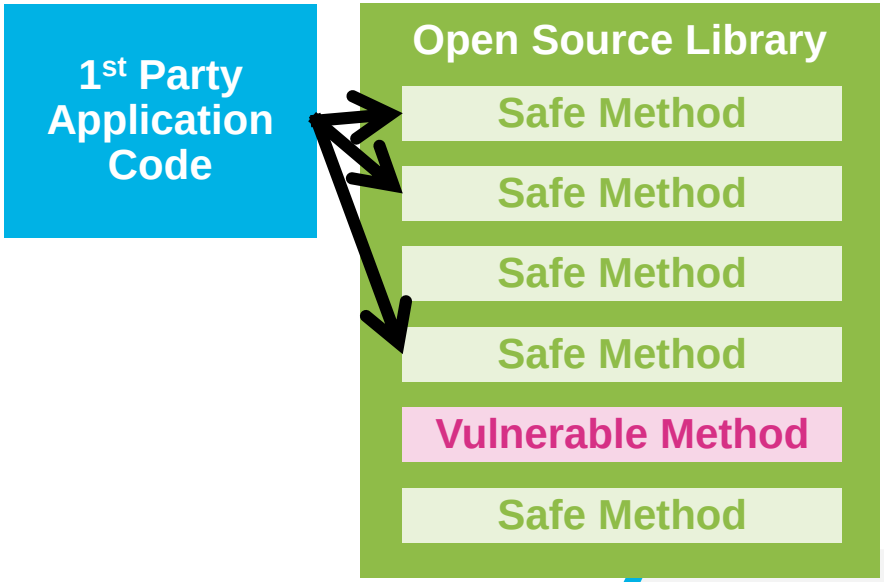

- OSS development embrace DevOps
  - NVD cannot cope with velocity and volume of submissions
- NVD CVEs do not provide exact library, vulnerable versions, and vulnerable code segment
- Hackers are watching OSS commits for silent fixes of vulnerabilities they can exploit in the wild

# Not Every OSS Vulnerability Is Exploitable

Risk prioritization

- Each OSS library may have 100s of functions and methods
- Vulnerability are usually only tied to one of these
- Your first party code only calls only a handful of methods in the library
- Most solutions don't allow you to prioritize applications where the vulnerable function is being called

Vulnerable Library?!



# Security Data is Not Equal

## CVE - Complete and Verified Public Data

- Verified Issues
- Vulnerable Versions & Safe Versions
- Vulnerable Methods
- Verify Fix

## SVE - Very Large Pool of Hidden Data

- Data-Mining of Upstream Open-Source
- Machine Learning + Human Verification

[:]SourceClear

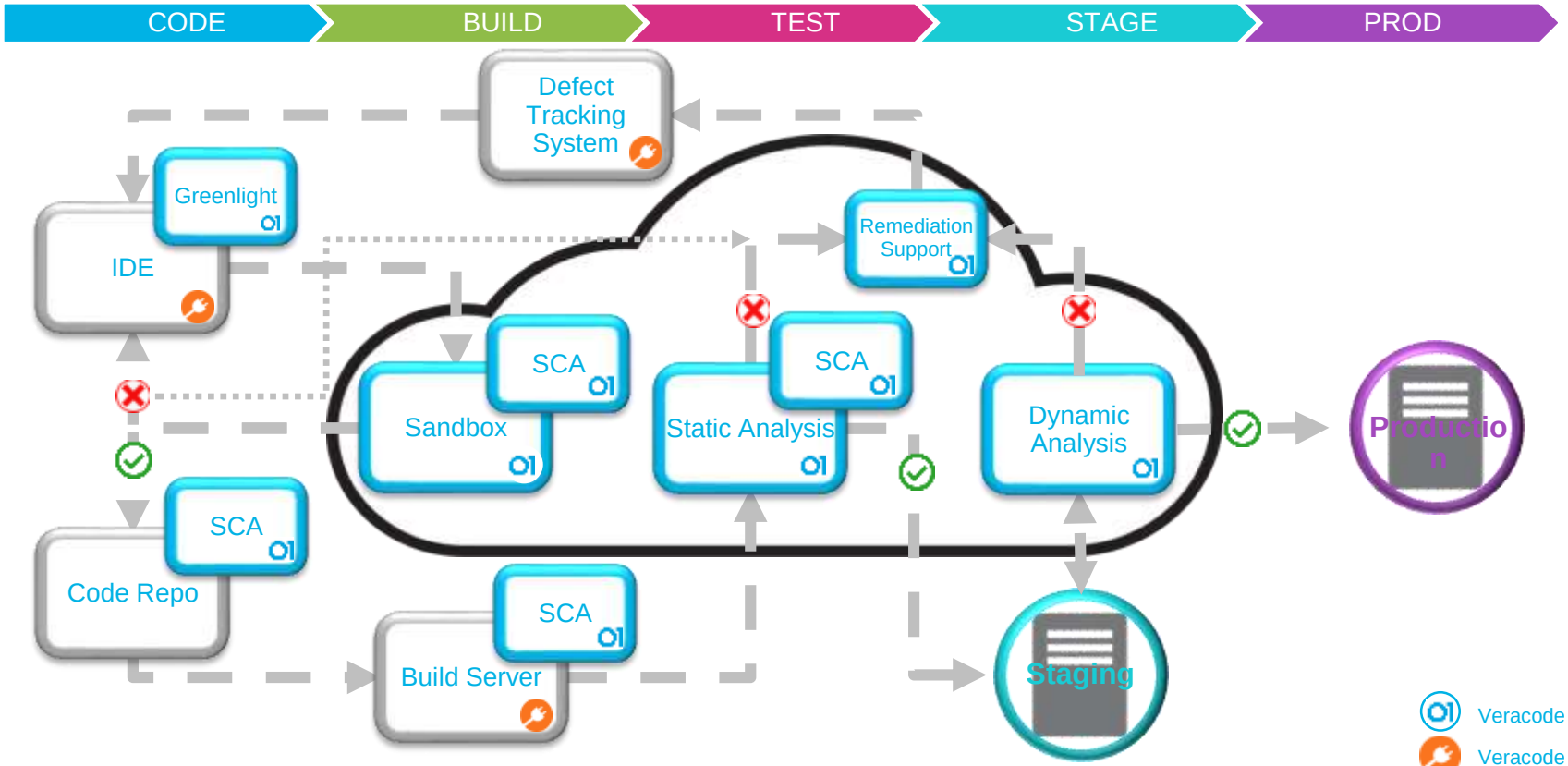
# Supported integrations / languages

| Build Tools | Languages | Vulnerable Methods |
|---|---|---|
| Bamboo | Java | yes |
| circleci | JavaScript | Q4 2019 |
| CODESHIP | .NET | yes |
| GitLab | Scala | |
| Travis CI | Go | H1 2020 |
| gradle | Python | yes |
| Bitbucket | Objective-C | |
| Jenkins | Ruby | yes |
| maven | PHP | |
| Command Line Integration | C/C++ | H1 2020 |

# Scanning Containers for Vulnerabilities

- SCA agent scans Docker containers and Docker Images
  - Scan can be part of CI script with docker build
- Agent detects packages in Docker file (yum/rpm)
- Identify open source vulnerabilities and license results on containers including OS-base libraries
- Docker on CentOS and RHEL Linux distributions
- Rules (policies) can break the build
- Only direct graph, no dependency graph or vulnerable methods on phase 1

# Automate Security into Existing SDLC



# Thank you

VERACODE

You change the world, we'll secure it.

**Julian Totzek-Hallhuber**
Principal Solution Architect

**O** +49.6128.9371777 **M** +49.160.97285004
**Email:** jtotzekhallhuber@veracode.com
**LinkedIn:** http://de.linkedin.com/pub/julian-totzek-hallhuber/0/738/b65/
**Twitter:** @d3v_rand0m