DEVSECOPS:

CHALLENGES & OPPORTUNITIES

Mohan Yelnadu

Builder turned Breaker turned Defender(DevSecOps)

Global Head of Application Security

Prudential Group

DevSecOps Live – Online Meetup

Date: March 16, 2020

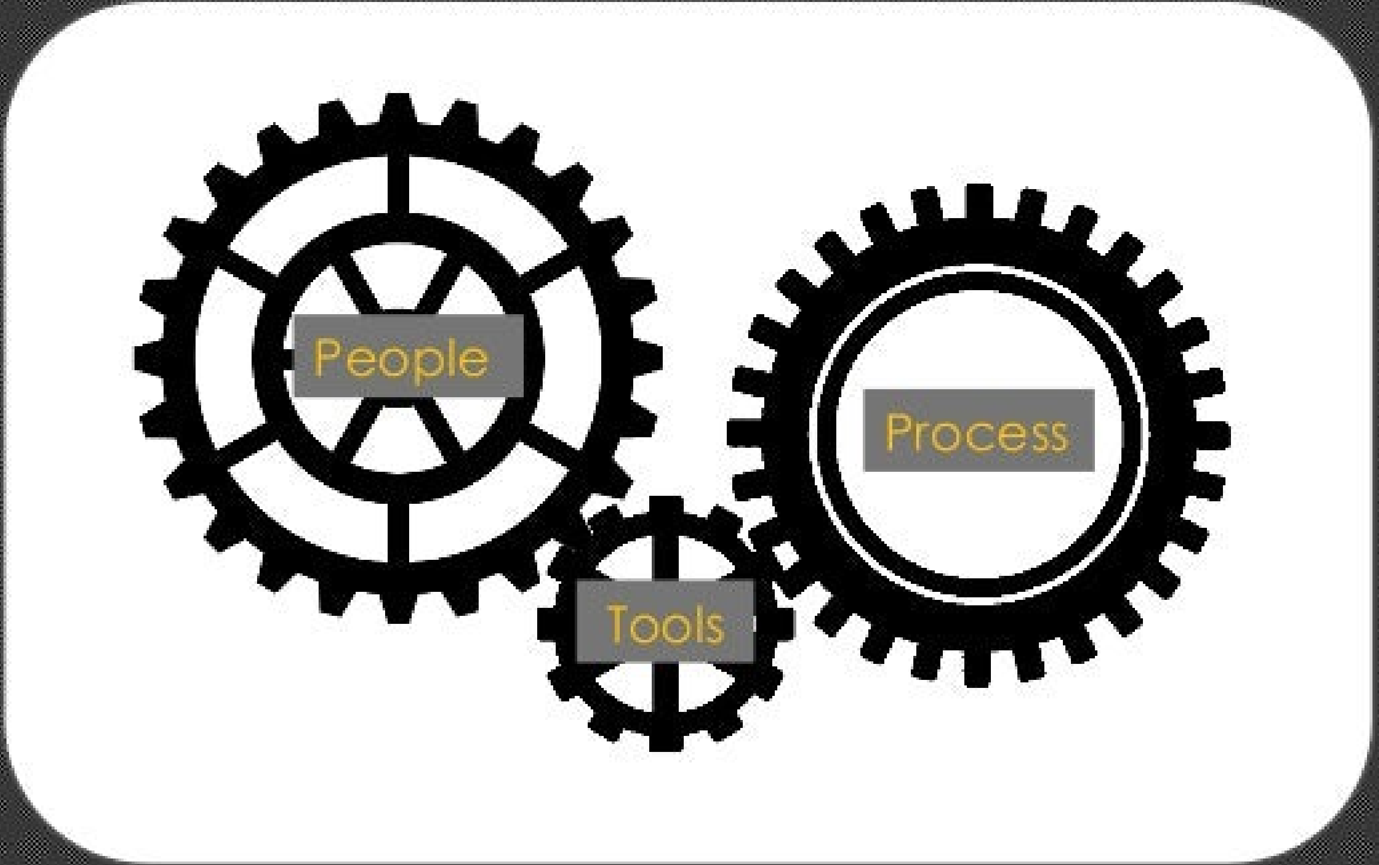
MOHAN YELNADU

APPLICATION SECURITY ARTIST

- Love traveling, understanding new Cultures
- Taught at University for 6 years
- Developed - Kernel Module, Ported Device Drivers
- Worked on - Static Analysis Compiler from Stanford Univ.
- Pentested - Web, Mobile and Infrastructure systems + IoT
- Banking, Insurance, Telecom, Manufacturing, Retail industries, across multiple geographies
- @monkelephant
- https://www.linkedin.com/in/mohanyelnadu
- Certs: OSCP, SABSA SCF

CHALLENGES: P. P. T

- People
  - Quality DevSecOps Consultants
  - Pragmatic Leadership
- Process
  - Open
  - Accommodating
- Tools
  - Right Set of DevSecOps Tools
  - To suit my Org Requirements



CHALLENGES: **PEOPLE**

- Investing in a **small but SMART** DevSecOps team is very critical

- A cohesive team **can achieve a lot** through automation
  - Application Security
  - Programming/Scripting
  - API Integration
  - Program Management
  - Stakeholder Management

- **Savvy** Leadership is key to program success
  - Change the language to talk to them
  - Articulate in terms of Business Risks

CHALLENGES: **APPSEC TOOLS**

- Choice of tools – very crucial
  - Can **make or break** your DevSecOps program
  - Decision makers – **CoE, Team with Curiosity..**
- To-Go or Not-To-Go by Industry Standard Reports
  - Get the direction
  - Do some research,
  - List your requirements,
  - **Go for PoC/Listen to Experts in the field**

CHALLENGES: **APPSEC TOOL GUIDANCE**

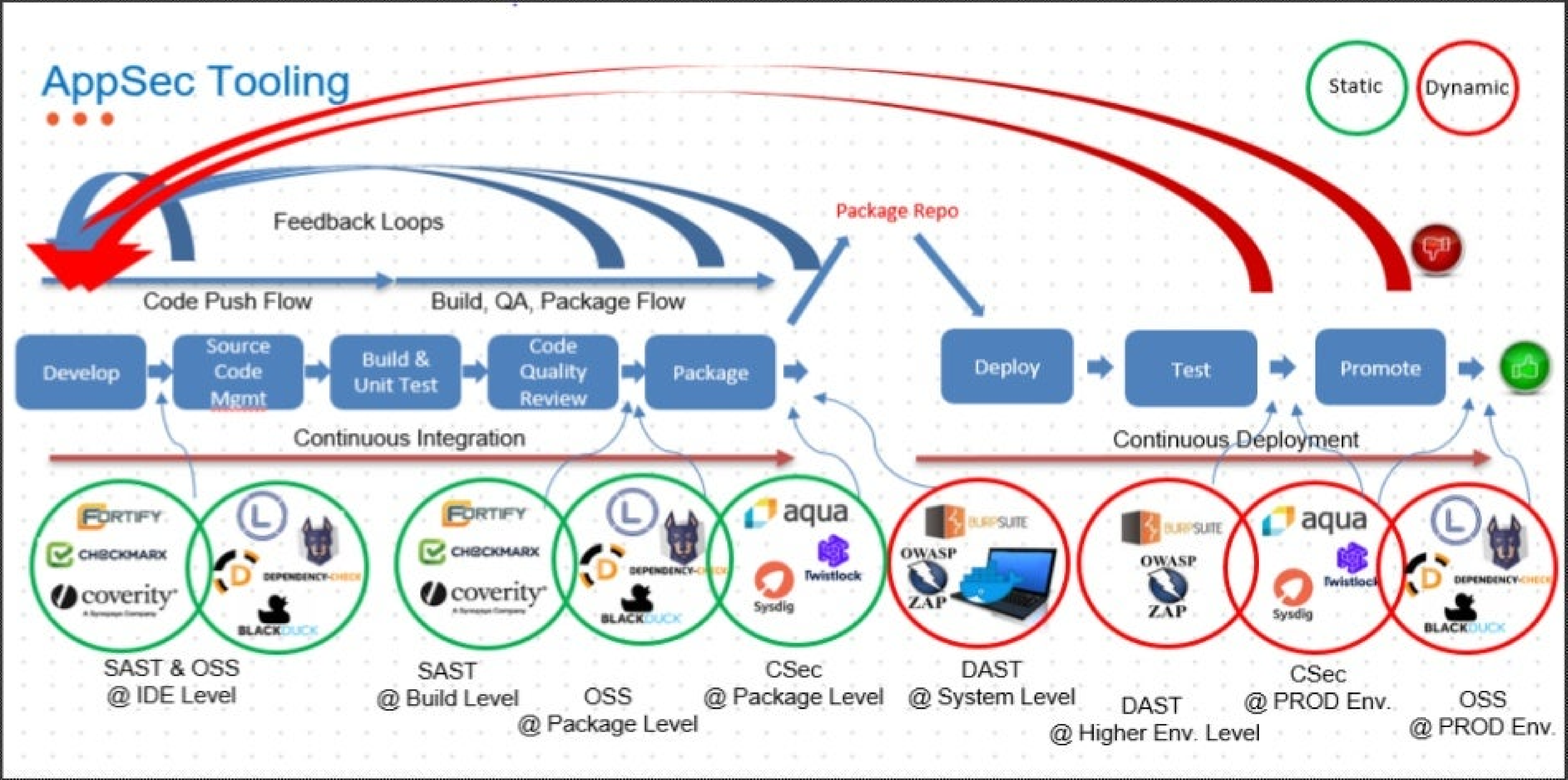
- Requirements:
  - Budget
  - Timeline
  - Ease of installation
  - Ease of use for
    - AppSec consultants
    - Developers
  - scanning Time (in day-to-day life)
  - Quality of Scanning-Feedback
    - On vulnerability
    - On mitigation/Recommendation

- Intuitive User Interface
- Ease of Metrics Management
- Support for APIs
- Integration with Secret Management Tools
- Ease of Integration with org process
- Vendor readiness to accommodate custom requests
- Product Roadmap
- Integration with Org CI/CD Ecosystem
- Out-of-box Compliance & other Reports

CHALLENGES: **PROCESS**

- Understand the existing process (Waterfall, Agile, CI/CD, etc.)
- Explore how effectively Developers can get feedback
- Identify at which dev stage, tool needs to be integrated
- how developers can start using it A**S**AP
  - **S**oon
  - **S**eamlessly
- Always remember: the motto
  - **Early, effortless, and constant feedback**

PROCESS: **EXAMPLE**



AppSec Tooling

Static

Dynamic

Feedback Loops

Package Repo

Code Push Flow

Build, QA, Package Flow

Develop → Source Code Mgmt → Build & Unit Test → Code Quality Review → Package → Deploy → Test → Promote

Continuous Integration

Continuous Deployment

SAST & OSS @ IDE Level

SAST @ Build Level

OSS @ Package Level

CSec @ Package Level

DAST @ System Level

DAST @ Higher Env. Level

CSec @ PROD Env.

OSS @ PROD Env.

SAST – Static Application Security Testing **OSS** – Open Source Software Security **CSec** – Container Security **DAST** – Dynamic Application Security Testing

OPPORTUNITIES / BEST PRACTICES

- Smooth onboarding
- Automate what you can
- Improving tool adoption
- Rollout Strategy
- Managing critical issues
- Making it work for SOC

- Production monitoring
- Tailored configuration
- Do the right thing
- Pragmatic Hygiene
- Managing zero days

SMOOTH ONBOARDING

- **Automated Onboarding on Security Tools**
  - Developers
  - Crucial: Time between
    - heard the tool name, & Onboarded

AUTOMATE WHAT YOU CAN

ONBOARDING

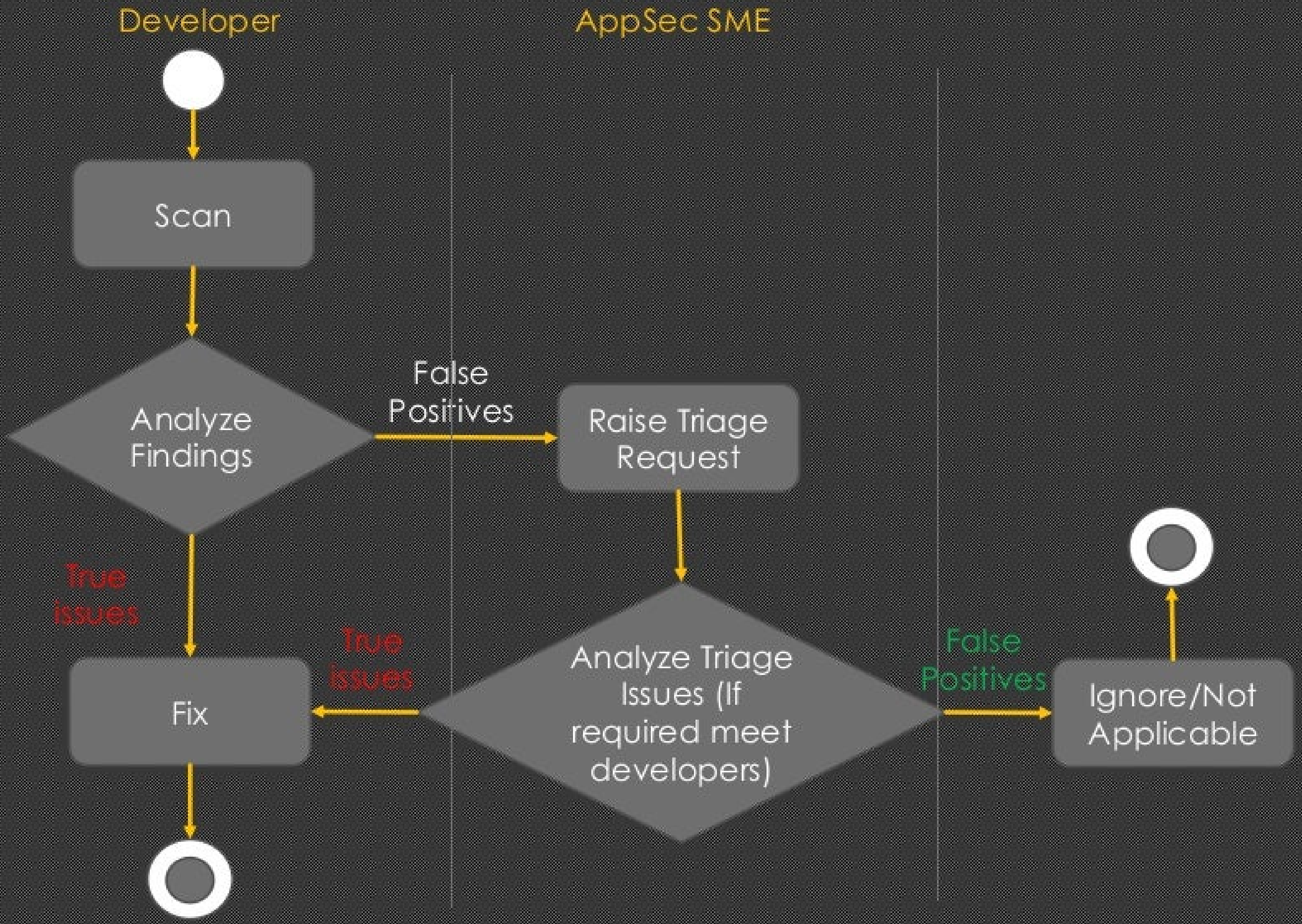
SCANNING

TRIAGE

BUILDBREAKER

ISSUE MANAGEMENT

TRIAGE



Developer

AppSec SME

Scan

Analyze Findings

False Positives

Raise Triage Request

True issues

True issues

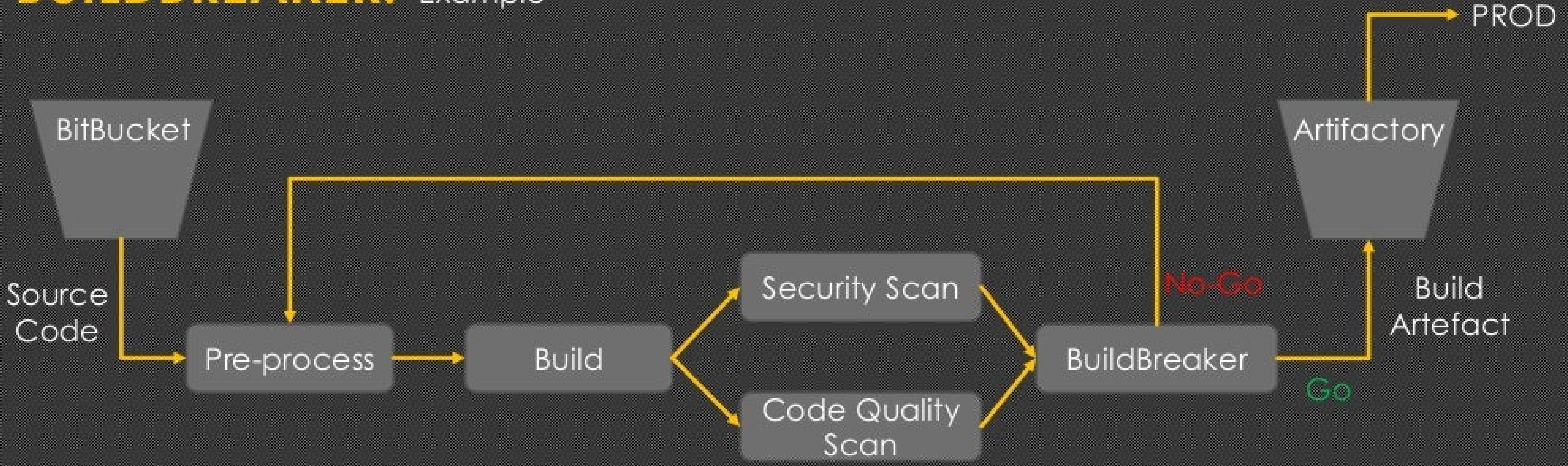
Fix

Analyze Triage Issues (If required meet developers)

False Positives

Ignore/Not Applicable

BUILDBREAKER: Example



BuildBreaker Example:

- No critical security issues in production build

IMPROVING TOOL ADOPTION

“ Give as many Live Demos as possible, share about new Tools & Processes ”

“ Allow developers to Get used to the Tools ”

“ Give enough notice while enabling BuildBreakers/Gating ”

“ Create Ecosystem: FAQs, Documentation, Demos, Videos ”

ROLLOUT STRATEGY

“ Break Build: In Stages ”

“ Handholding in False Positive Analysis: Triage & Guidance ”

“ Dispensation Management: Logging & Validity ”

MANAGING CRITICAL ISSUES

- Critical Issue Management
  - Level 10/Critical
    - Identification
    - Remediation
    - Follow-up
    - Closure

> “Developers DO NOT realise the Gravity of Level 10 OSS Issues
>
> Self-Expérience
> ☺”

WORKING WITH SOC

- Reaching Out to Right Stakeholders
  - SOC – access to DevSecOps Dashboard
  - Need to identify right stakeholder
  - **Mapping App with Right Stakeholders in Dashboard**

PRODUCTION MONITORING

- **Monitoring** Production Artefacts
  - Tools have these features, enable it
  - Apply **nightly** check
  - Enable **alert** option

“ Effective PROD Monitoring saved a huge effort!

Self-Expérience ☺ ”

TAILORED CONFIGURATION

- Custom Roles
  - Pre-defined roles are good, sometimes need more
  - Create Custom Role
  - Ex: Developers could allow **Dispensation** to themselves in the tool
    - **Logging** helped to trace out
  - **Create Developer** Role and give min (Scan, View, etc.) privileges

DO THE RIGHT THING

- Educate developers to use Local Scans
  - Use Tool dashboard to **Upload Library and Analyse**
  - Use tool's **browser plugin to scan**
  - Use **IDE Plugin to enable local scans**

PRAGMATIC HYGIENE

- KEEP **UPGRADING THE TOOLS TO LATEST VERSIONS**

- VENDORS BRING **NEW FEATURES**, **INNOVATIONS** REGULARLY

- **ANALYSE** NEW FEATURES **IN TEST ENVIRONMENT**, THEN ROLL OUT

MANAGING ZERO DAYS

- Keep **eyes and ears open** to sense any **zero days:**
  - either in **your libraries** or **tools**
- Prepare a plan to act in such situations
  - **WAF** readiness
  - **Constant touch with vendor** representatives
  - Right stakeholders **EVER ready to act**

IMPORTANT : SECRETS MANAGEMENT

- Do Not store your secrets in Plain Text
  - Source Code, Pipelines, DB configurations, Application Layer Secrets
- Use dedicated secret management tools
  - CyberArk, Hashicorp Vault, AKV, etc.

THANK YOU!

Mohan Yelnadu

Builder-turned-Breaker-turned-Defender(DevSecOps)

Head, Application Security

Prudential Group

@monkelephant

https://www.linkedin.com/in/mohanyelnadu