ASMGi

How Do I "DevSecOps"?

Presented by ASMGi

October 27, 2020

Today’s Presenters – ***How Do I “DevSecOps”?***

John Meyer

Solutions Architect, ASMGi

jmeyer@asmgi.com

Frank Yako

CIO, Director of Strategic Initiatives

fyako@asmgi.com

Agenda

ASMGi

- Introduction and Objectives
- Security at Every Level
- Perform a Security / Risk Assessment
- Automate
- Dashboards and Alerts for Continuous Monitoring
- Be in a Continuous State of Compliance

Introduction and Objectives

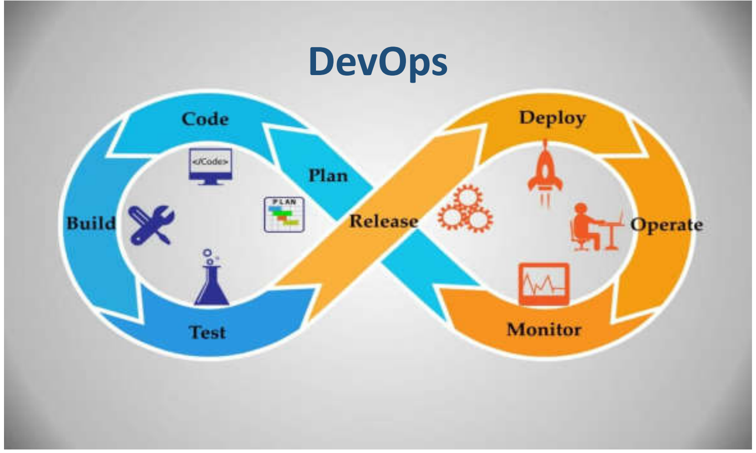
You’ve learned how to assess your current maturity in the DevSecOps model. So what’s next? How do you implement a mature DevSecOps program? This is what we mean by ‘How do I DevSecOps’!

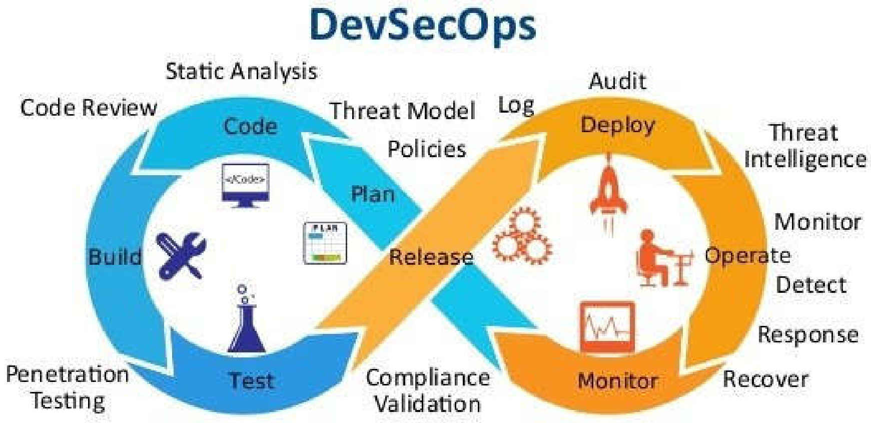
- Discuss **Methods** to implement a DevSecOps Program
- Learn how to implement and integrate **Security Tools** and **Processes** throughout the SDLC
- The importance of **Automation** in every process possible
- The importance of **Continuous Monitoring**
- How to maintain **Continuous Compliance** through the DevSecOps model

DevSecOps – Security at Every Level

It Starts…and Ends with Security

- “You have to know where you’re at to know where your going”
- And it starts with SECURITY
- Security integrated into every phase of the DevOps Lifecycle





DevSecOps – Where to Begin???

Self-Assessment – It starts with Security

- “You have to know where you’re at to know how to reach the destination”.
- Complete an Assessment and determine what’s needed to create a successful DevSecOps Program.
- With the Assessment complete it’s time to Prioritize and Remediate!

A Method to the Model

- Evaluate Roles, Processes, and Tools
- Does each Role, Process, and/or Tool serve a purpose and bring value?
- Are there any Bottlenecks?
- Are there any missing Roles, Processes, or Tools?
- Design for an outcome!

Source: ScienceSoft 2020

Choosing the Right _[blank]_ for the Job

Choosing and Implementing Processes and Tooling

- Threat Modeling
- Penetration Testing
- Code Scans
- Code Review
- License Compliance
- Open Source Compliance
- Hardened Container/OS Image
- WAF and Service Mesh

Listing Pros and Cons

RACI Diagrams

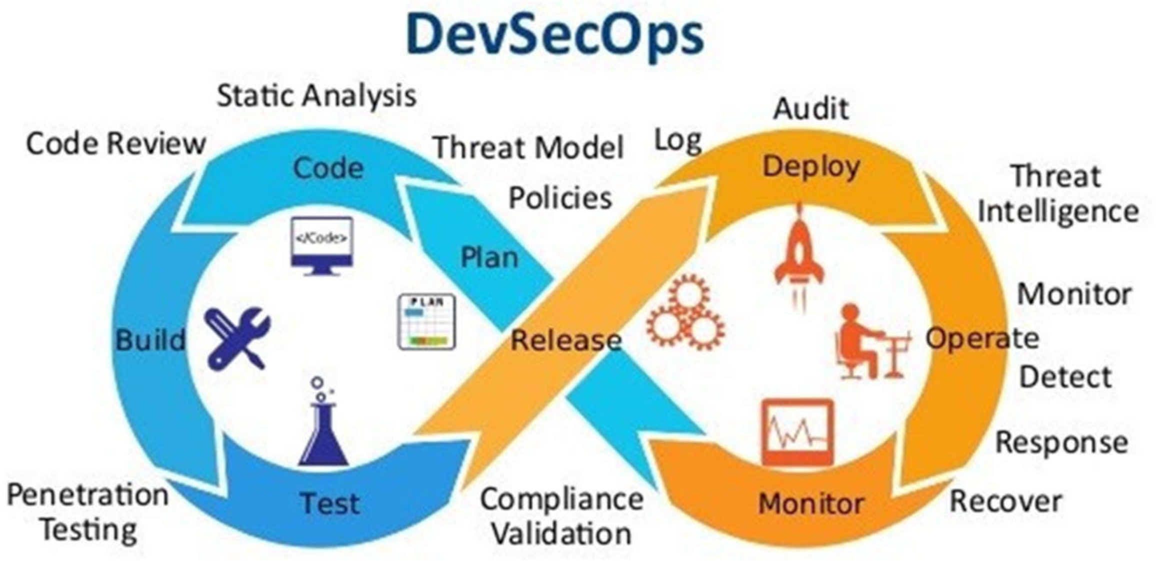
Cost Analysis

Risk Analysis

Security is Everyone's Responsibility

Empower developers to suggest critical security changes. Make everyone accountable for security empowering your teams with tools and expertise to respond to (and neutralize) threats before they become a major issue.

(Sumo Logic)

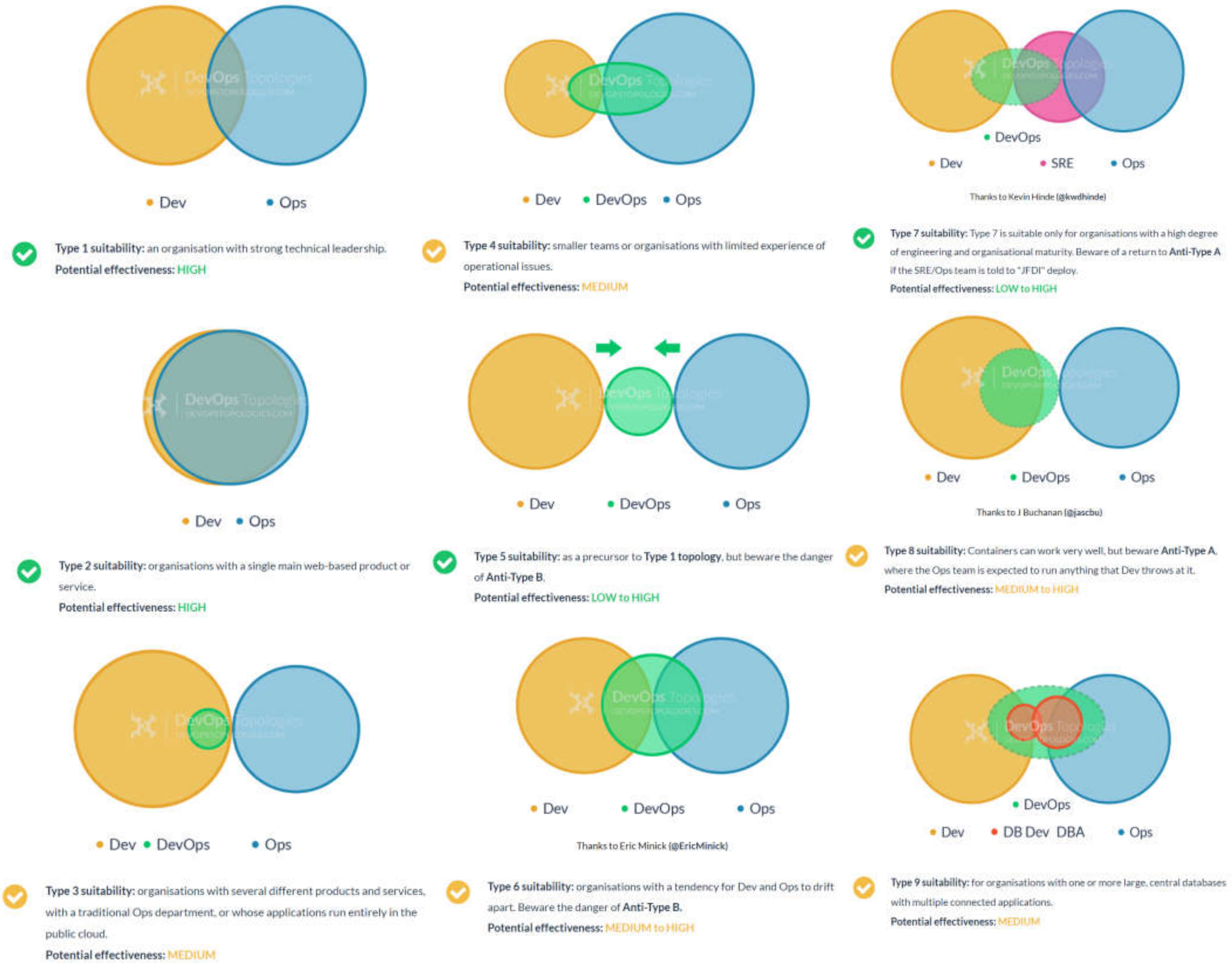


DevOps Program Methodologies

Not a One Size Fits All

- Figure out what DevOps model suits your organization or project need.
- DevSecOps Can be as broad as an organization or as granular as project based.

> The goal for a "DevOps Team" should be to put itself out of business by enabling the rest of the org.
>
> EricMinick



Type 1 suitability: an organisation with strong technical leadership.
Potential effectiveness: HIGH

Type 2 suitability: organisations with a single main web-based product or service.
Potential effectiveness: HIGH

Type 3 suitability: organisations with several different products and services, with a traditional Ops department, or whose applications run entirely in the public cloud.
Potential effectiveness: MEDIUM

Type 4 suitability: smaller teams or organisations with limited experience of operational issues.
Potential effectiveness: MEDIUM

Type 5 suitability: as a precursor to **Type 1 topology**, but beware the danger of **Anti-Type B**.
Potential effectiveness: LOW to HIGH

Thanks to Eric Minick (@EricMinick)

Type 6 suitability: organisations with a tendency for Dev and Ops to drift apart. Beware the danger of **Anti-Type B**.
Potential effectiveness: MEDIUM to HIGH

Thanks to Kevin Hinde (@kwdhinde)

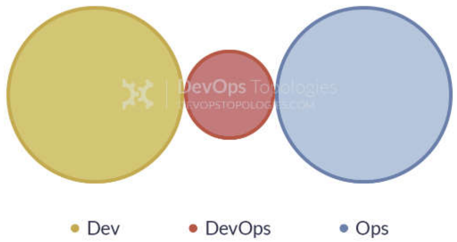
Type 7 suitability: Type 7 is suitable only for organisations with a high degree of engineering and organisational maturity. Beware of a return to **Anti-Type A** if the SRE/Ops team is told to "JFDI" deploy.
Potential effectiveness: LOW to HIGH

Thanks to J Buchanan (@jascbu)

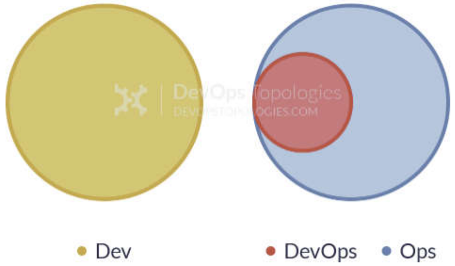
Type 8 suitability: Containers can work very well, but beware **Anti-Type A**, where the Ops team is expected to run anything that Dev throws at it.
Potential effectiveness: MEDIUM to HIGH

Type 9 suitability: for organisations with one or more large, central databases with multiple connected applications.
Potential effectiveness: MEDIUM

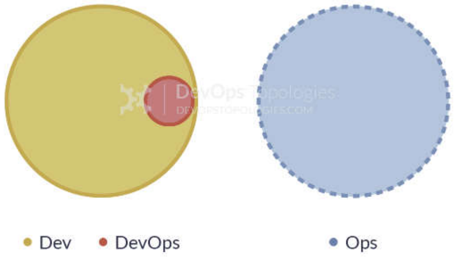
Methods to Avoid



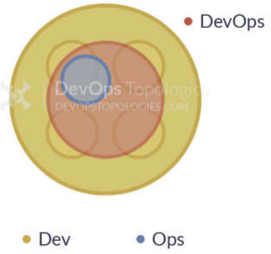
Anti-Type B: DevOps Team Silo



Anti-Type E: Rebranded SysAdmin

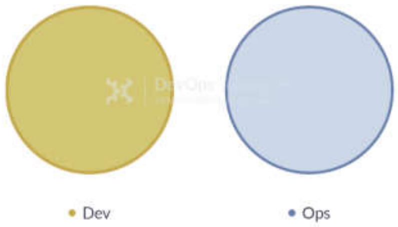


Anti-Type C: Dev Don't Need Ops

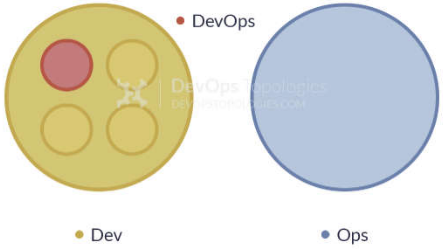


Thanks to Matt Franz (@seclectech)

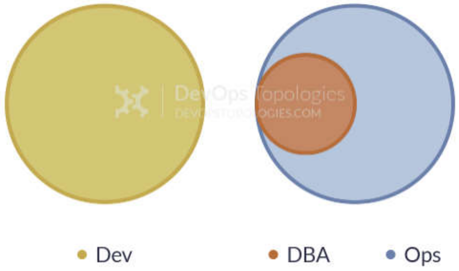
Anti-Type F: Ops Embedded in Dev Team



Anti-Type A: Dev and Ops Silos



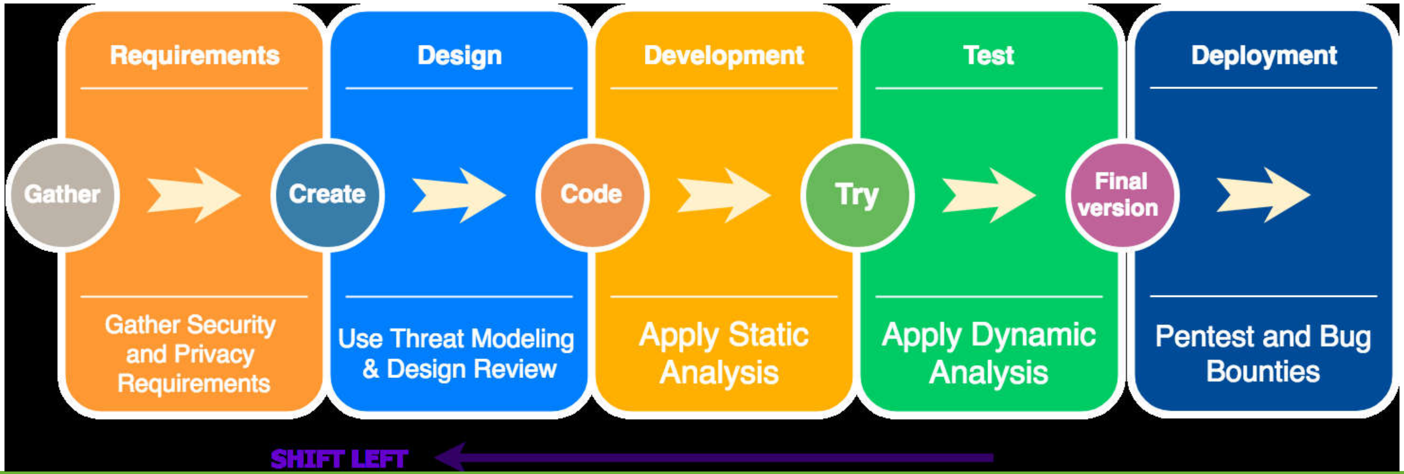
Anti-Type D: DevOps as Tools Team



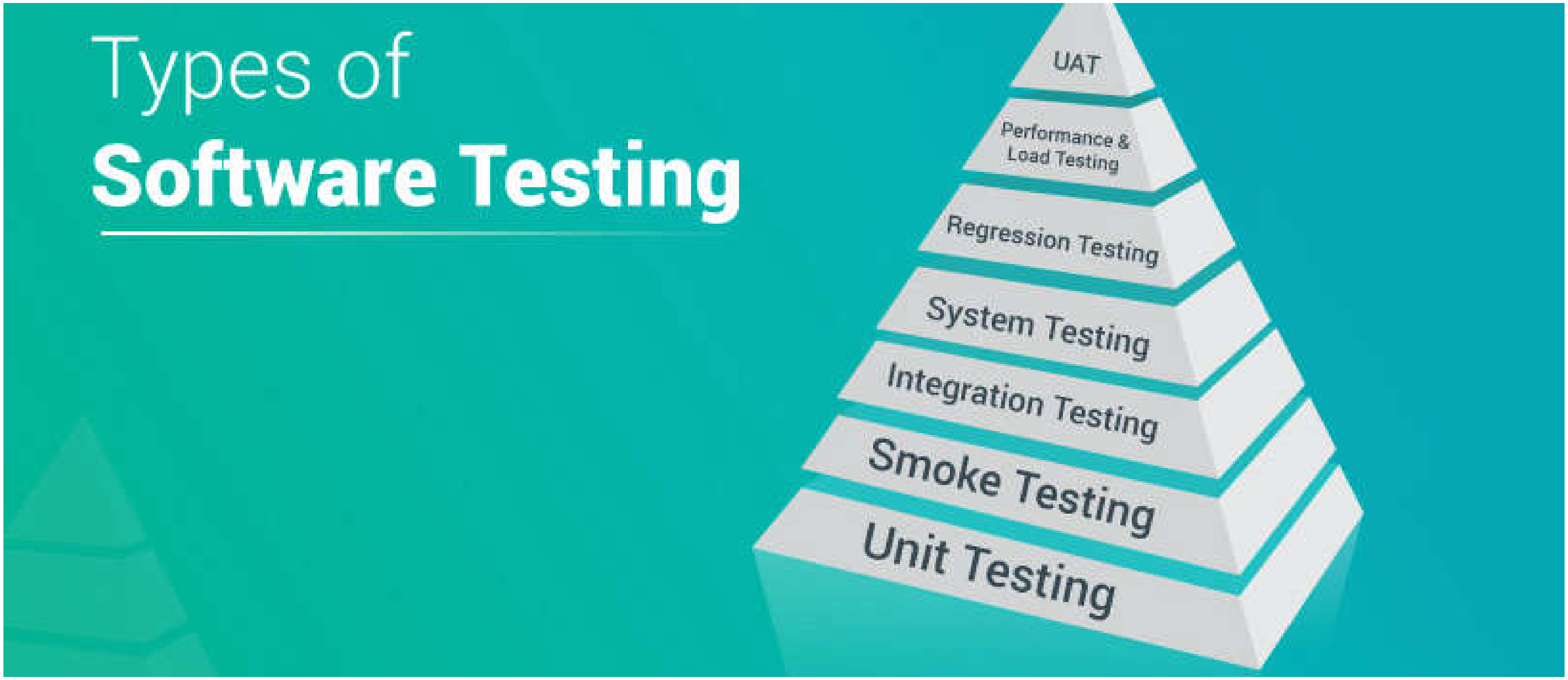
Anti-Type G: Dev and DBA Silos

Tools to Enable Secure Code

- SCA → Open Source Compliance
- SCA → License Compliance
- SAST → Source Code Scan
- DAST → Web Security Scanning
- IAST → Interactive Security Scanning
- IDE Plug-Ins → Real-Time Code Scanning
- CI/CD → Automated Testing



Software Testing Methods

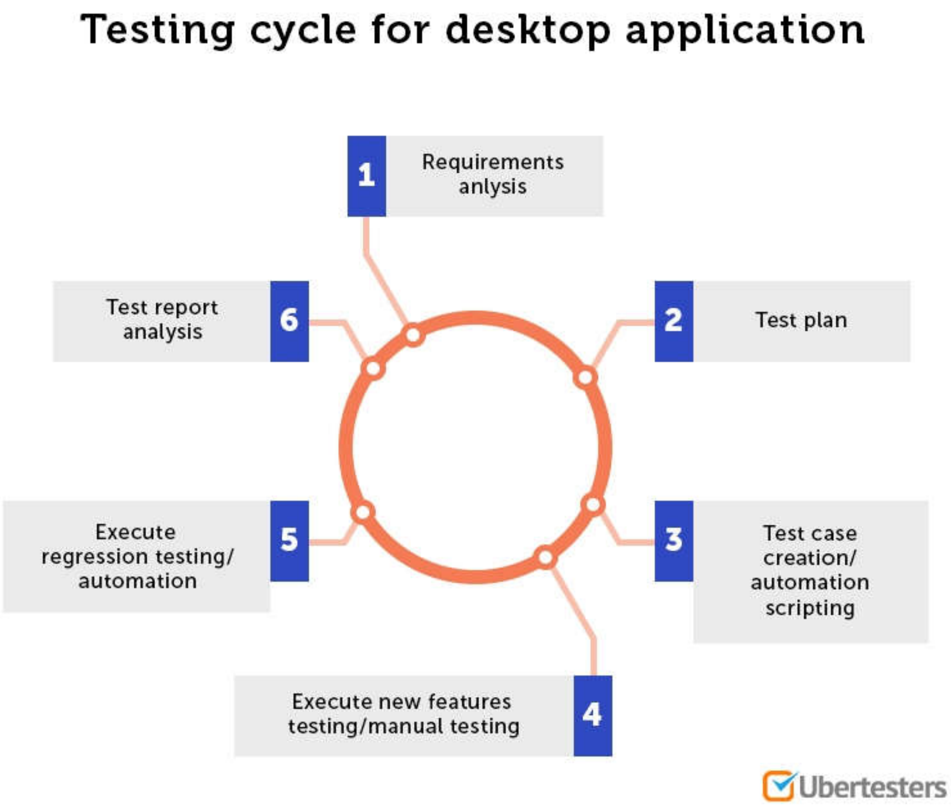


Test Automation for Secure Code and Secure Applications

- Why is automation important in software testing?

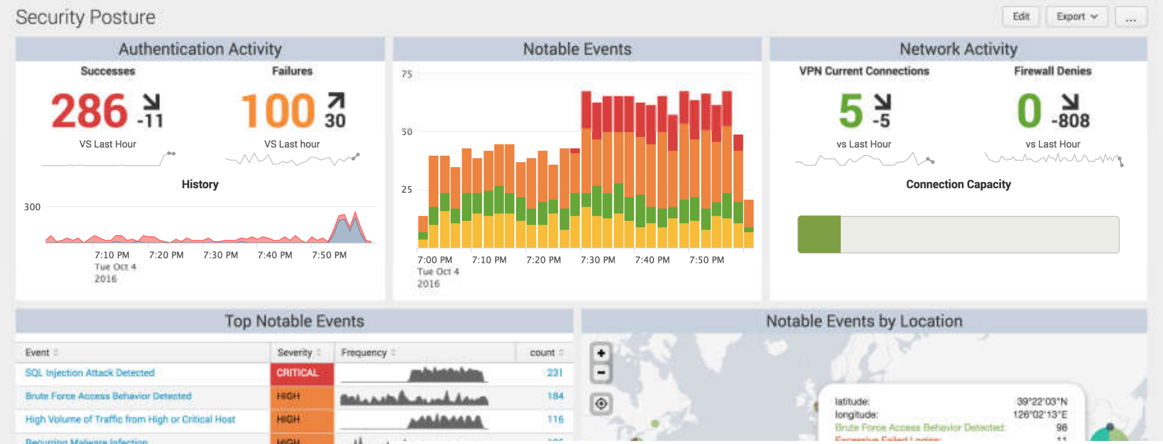
  Automated software testing can increase the depth and scope of **tests** to help improve **software** quality. ... **Test automation** can easily execute thousands of different complex **test** cases during every **test** run providing coverage that is impossible with **manual tests**.

- Benefits of Automated Testing
  - Faster Feedback Cycle. Without **test automation**, feedback for newly developed features can take a while. ...
  - Team Saves Time. ...
  - Reduced Business Expenses. ...
  - Higher **Test** Coverage. ...
  - Reusability of **Test** Suite. ...
  - Faster Time to Market. ...
  - Better Insights. ...
  - Improved Accuracy
  - Automated Testing Provides More Features
  - Less Stress on QA Team
  - Quickly Determine the Stability of Your Build
  - Eliminate Human Error


Testing cycle for desktop application
1 Requirements anlysis
2 Test plan
3 Test case creation/ automation scripting
4 Execute new features testing/manual testing
5 Execute regression testing/ automation
6 Test report analysis
Ubertesters

Continuous Monitoring, Alerting, & Dashboards

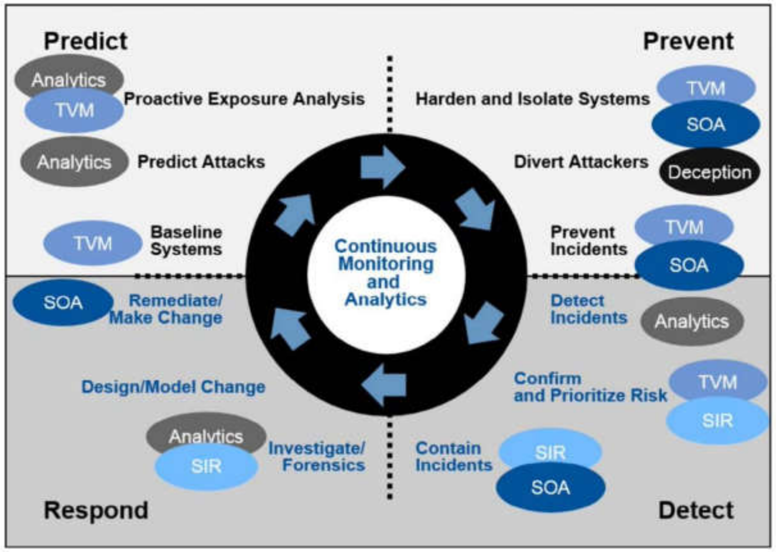
- ◆ Effective Monitoring
  - Systems
  - Application
  - Infrastructure
  - Security
- ◆ Meaningful Alerts
- ◆ Accurate Reporting
- ◆ Risk Assessment



Top Notable Events

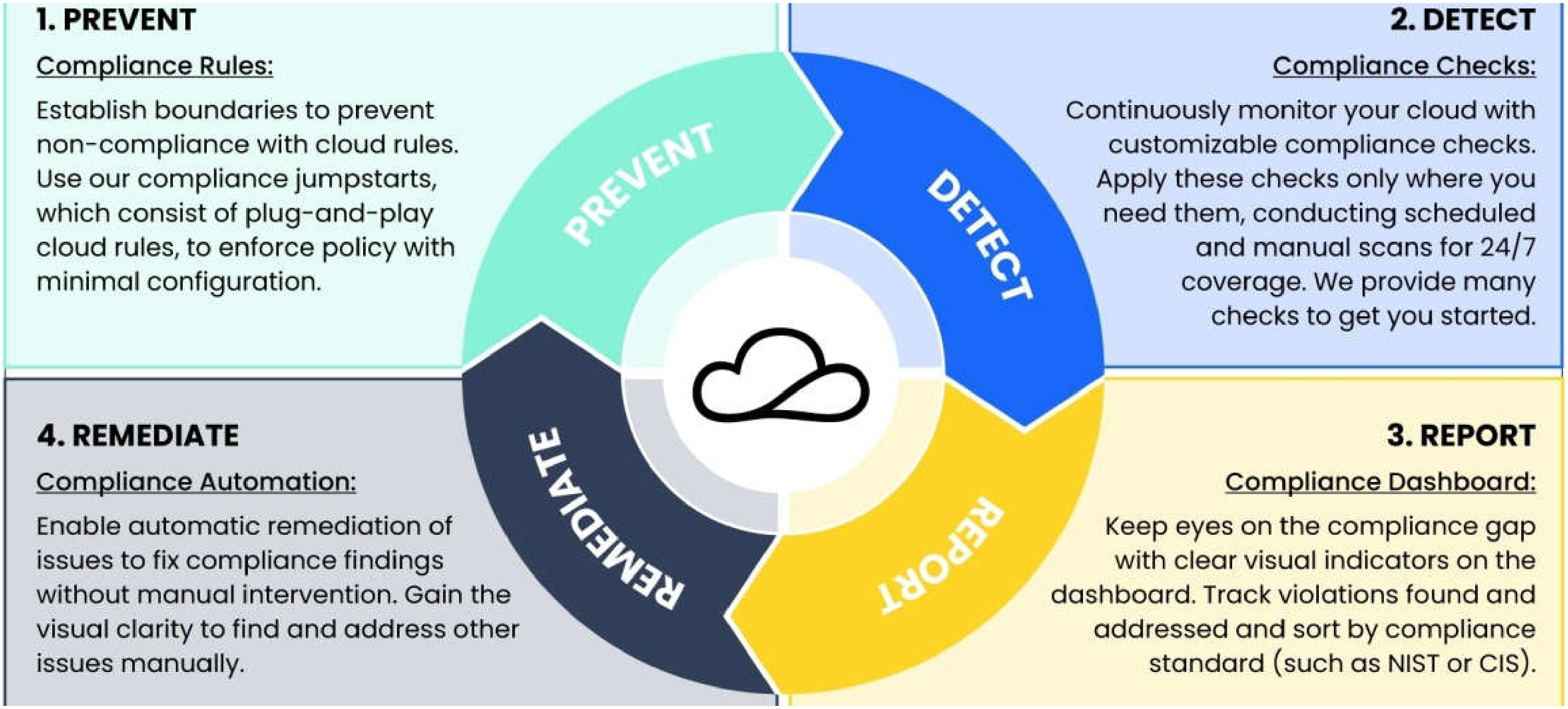
Event	Severity	Frequency	count
SQL Injection Attack Detected	CRITICAL		231
Brute Force Access Behavior Detected	HIGH		184
High Volume of Traffic from High or Critical Host	HIGH		116
Recurring Malware Infection	HIGH		
Expected Host Not Reporting	LOW		
Insecure Or Cleartext Authentication	MEDIUM		
Activity from Expired User Identity	MEDIUM		
Excessive Failed Logins	HIGH		
Host With Multiple Infections	HIGH		
Anomalous Audit Trail Activity Detected	LOW		

latitude:	39°22'03"N
longitude:	126°02'13"E
Brute Force Access Behavior Detected:	98



Source: Gartner (November 2015)

Continuous Compliance



1. PREVENT

Compliance Rules:

Establish boundaries to prevent non-compliance with cloud rules. Use our compliance jumpstarts, which consist of plug-and-play cloud rules, to enforce policy with minimal configuration.

2. DETECT

Compliance Checks:

Continuously monitor your cloud with customizable compliance checks. Apply these checks only where you need them, conducting scheduled and manual scans for 24/7 coverage. We provide many checks to get you started.

3. REPORT

Compliance Dashboard:

Keep eyes on the compliance gap with clear visual indicators on the dashboard. Track violations found and addressed and sort by compliance standard (such as NIST or CIS).

4. REMEDIATE

Compliance Automation:

Enable automatic remediation of issues to fix compliance findings without manual intervention. Gain the visual clarity to find and address other issues manually.

(Graphic: Business Wire)

…What’s Next

Join us for the next session to apply priorities identified through a DevSecOps Assessment…

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

ASMGi

Thank You!

800 Superior Ave E, Ste 1050
Cleveland, OH 44114

Phone: 216.255.3040
Fax: 216.274.9647

Email: info@asmgi.com

www.asmgi.com