Shift-Left Security with the Security Test Pyramid

Andreas Falk
@andifalk
https://github.com/andifalk
https://www.novatec-gmbh.de/en

I am we are.

devoxx uk
the developer community conference

About Me

Andreas Falk

Novatec Consulting (Germany)

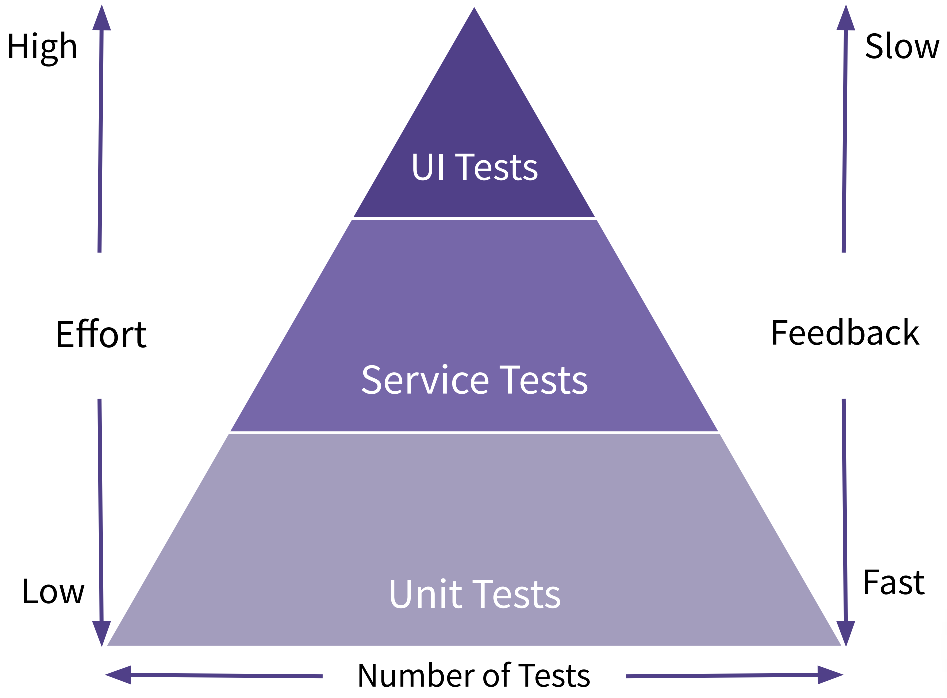
andreas.falk@novatec-gmbh.de

@andifalk

OWASP Open Web Application Security Project

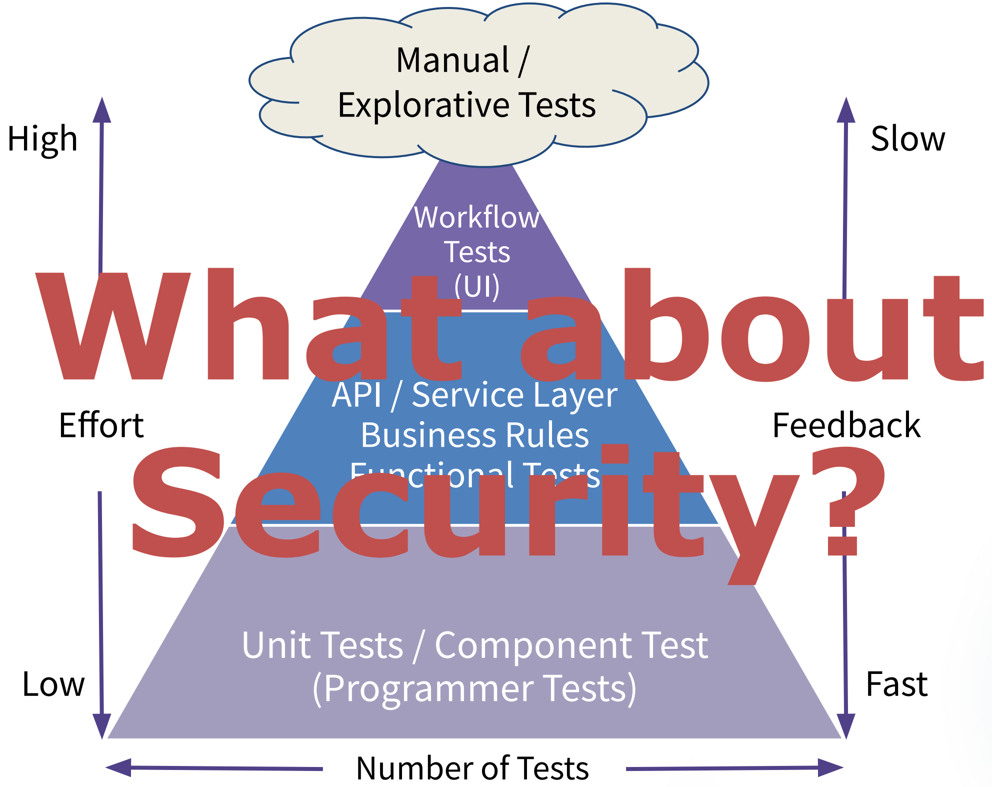
OpenID

The Testing Pyramid (by Mike Cohn)



https://www.mountaingoatsoftware.com/blog/the-forgotten-layer-of-the-test-automation-pyramid
https://martinfowler.com/articles/practical-test-pyramid.html

The Testing Pyramid (by Gregory / Crispin)


Manual / Explorative Tests
High
Slow
Workflow Tests (UI)
What about Security?
API / Service Layer Business Rules Functional Tests
Effort
Feedback
Unit Tests / Component Test (Programmer Tests)
Low
Fast
Number of Tests

https://agiletester.ca/more-agile-testing-the-book

OWASP Top 10 - 2021

NEW

OWASP Open Web Application Security Project

https://owasp.org/Top10

A01:2021-Broken Access Control

A02:2021-Cryptographic Failures

A03:2021-Injection

A04:2021-Insecure Design

A05:2021-Security Misconfiguration

A06:2021-Vulnerable and Outdated Components

A07:2021-Identification and Authentication Failures

A08:2021-Software and Data Integrity Failures

A09:2021-Security Logging and Monitoring Failures

A10:2021-Server-Side Request Forgery

DEVOXX United Kingdom

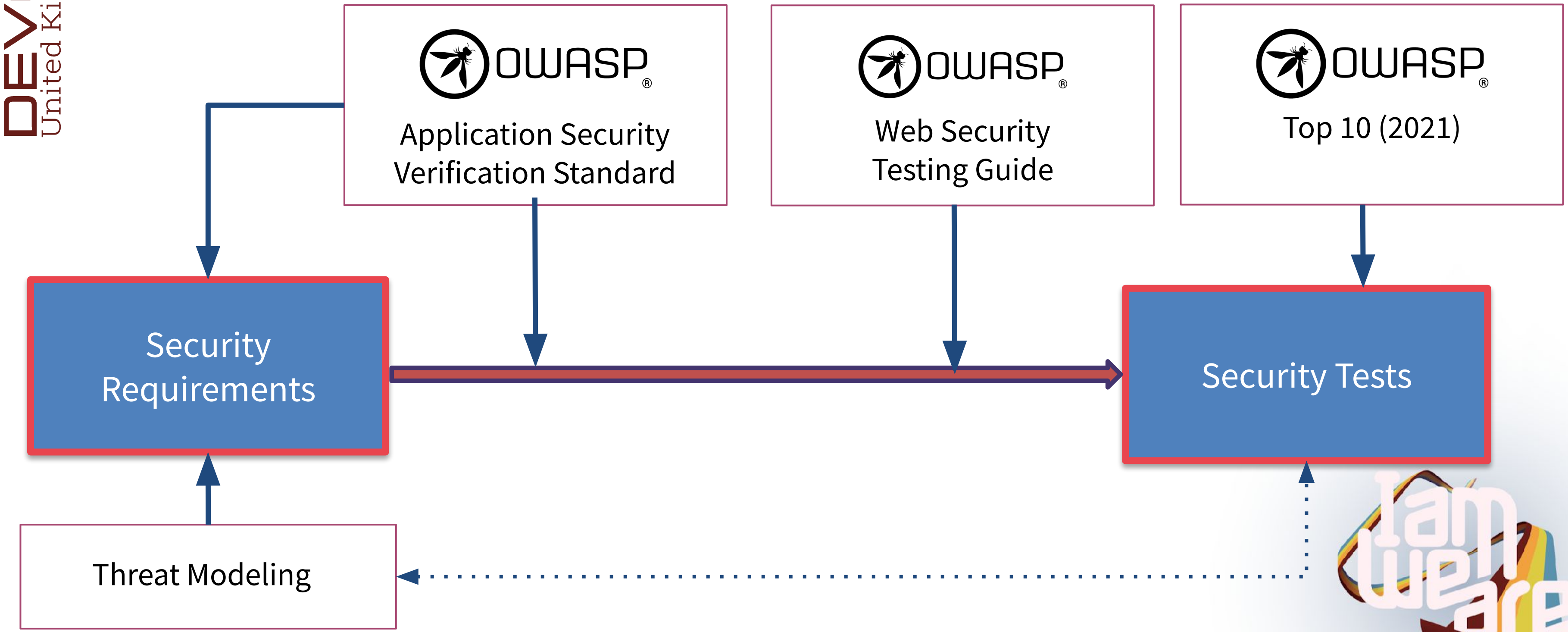
OWASP Top 10 - A04:2021 – Insecure Design

- Use **threat modeling** for critical authentication, access control, business logic, and key flows
- Integrate **security** language and controls into **user stories**
- Write **unit and integration tests** to validate that all critical flows are resistant to the threat model
- Compile **use-cases** and **misuse-cases** for each tier of your application
- …

https://owasp.org/Top10/A04_2021-Insecure_Design

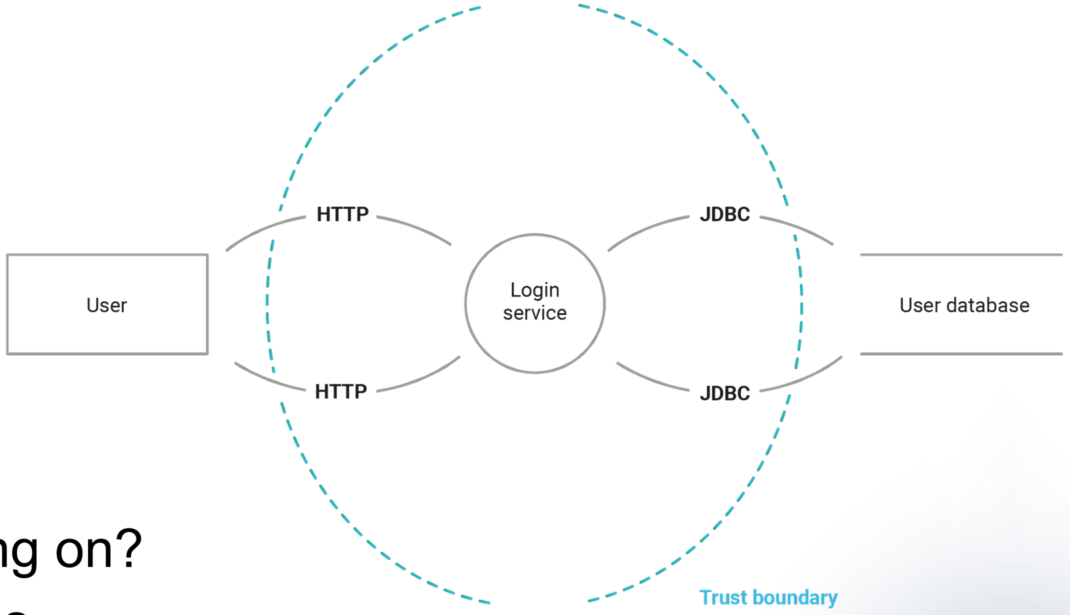
devoxx uk
the developer community conference

Derive Tests from Security Requirements



https://owasp.org/www-project-application-security-verification-standard
https://owasp.org/www-project-web-security-testing-guide

Threat Modeling



Four Key Questions:

1. What are we working on?
2. What can go wrong?
3. What are we going to do about it?
4. Did we do a good enough job?

https://www.threatmodelingmanifesto.org

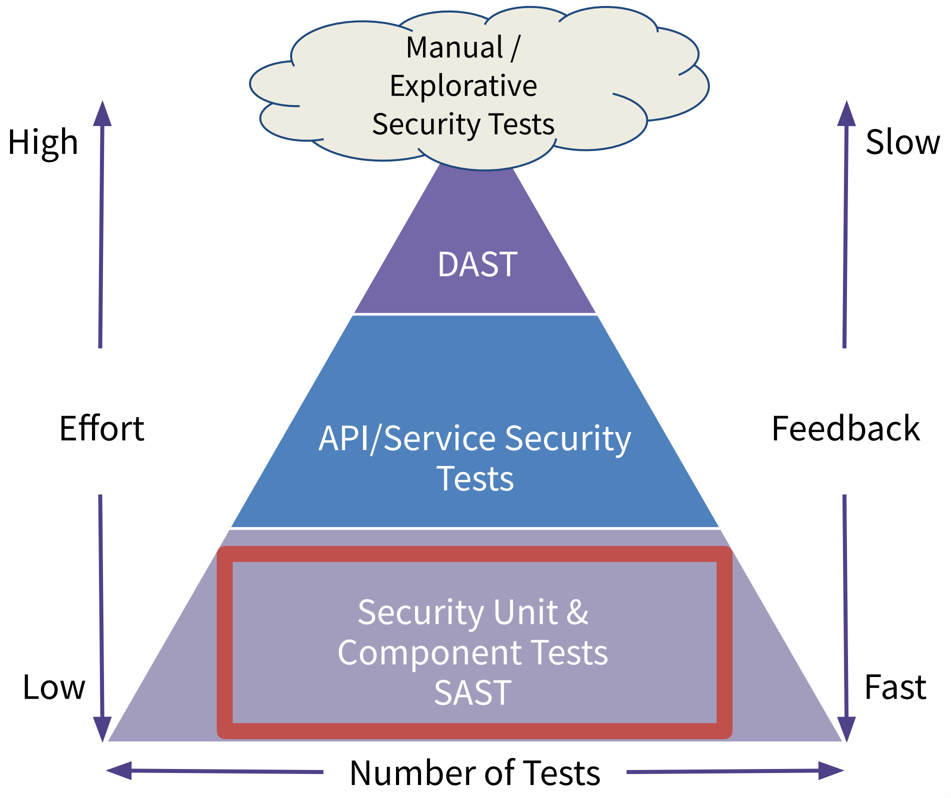
Application Security Verification Standard

	Applicability	Building			Building, Configuration, Deployment Assurance and Verification			Assurance and Verification	
Level 1	🟨 All apps		🟨 Secure Coding	🟨 Standards and checklists	🟨 Secure & Peer Code Review	🟨 DevSecOps	🟨 Unit and Integration Tests	🟩 Penetration Testing	🟨 DAST
Level 2	🟩 All apps	🟩 Security Architecture and Reviews	🟩 Secure Coding	🟩 Standards and checklists	🟩 Secure & Peer Code Review	🟩 DevSecOps	🟩 Unit and Integration Tests	🟩 Hybrid Reviews	🟨 SAST
Level 3	🟩 High Assurance	🟩 Security Architecture and Reviews	🟩 Secure Coding	🟩 Standards and checklists	🟩 Secure & Peer Code Review	🟩 DevSecOps	🟩 Unit and Integration Tests	🟩 Hybrid Reviews	🟨 SAST

Legend: 🟨 Acceptable 🟩 Suitable

https://github.com/owasp/asvs

The Security-Testing Pyramid: Unit Test Layer


Manual / Explorative Security Tests
DAST
API/Service Security Tests
Security Unit & Component Tests SAST
High
Effort
Low
Slow
Feedback
Fast
Number of Tests

- SAST
- No Known Vulnerabilities
  - Dependency Check
  - Container Image Scan
- Unit / Component Tests
  - Injection (Input Validation)
  - Broken Authentication
  - Bypass Business Logic
  - Errors & Logging
  - Secure Architecture

Security on the Unit Testing Layer

- Static Application Security Testing (SAST)
- Secure Architecture
- Broken Authentication (Password Policies)

Static Application Security Testing (SAST)

SAST solutions analyze an application from the "inside out" in a nonrunning state

Some SAST tools:

- SpotBugs (Java/Kotlin - with *Find Security Bugs* plugin)
- SonarQube
- SemGrep
- Snyk Code
- Checkov & Terrascan (Infrastructure as Code)
- Trufflehog (Check git repository for secrets)
- OWASP Dependency Check (Vulnerable Libs)

Testing Secure Architecture & Design

- Violations of Architecture Patterns (Layered, Onion)
- Broken Authentication (e.g. Password Policies)
- Missing Input Validation
- Broken Authorization (No Authz checks)
- Invalid calls to Persistence APIs

https://www.archunit.org

DEVOXX United Kingdom

V2: Authentication (ASVS)

Passay

- **V2.1 Password Security Requirements**
  - 2.1.1 Verify that user sets passwords are at least 12 characters in length
  - 2.1.2 Verify that passwords 64 characters or longer are permitted but may be no longer than 128 characters
  - 2.1.3 Verify that password truncation is not performed
  - 2.1.4 Verify that any printable Unicode character is permitted in passwords
  - 2.1.7 Verify that passwords are checked against a set of breached passwords

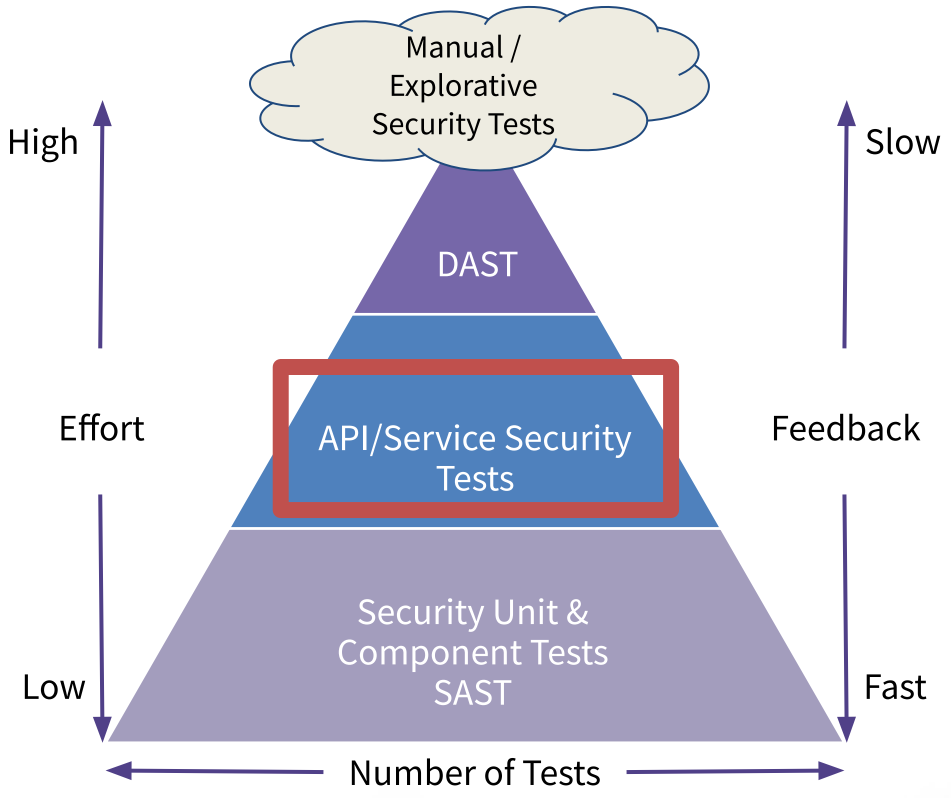
https://owasp.org/www-project-application-security-verification-standard
https://www.passay.org

Demo

Unit Test Layer

https://github.com/andifalk/bookmark-service

The Security-Testing Pyramid: Service Test Layer


Manual / Explorative Security Tests
DAST
API/Service Security Tests
Security Unit & Component Tests SAST
High
Effort
Low
Slow
Feedback
Fast
Number of Tests

- **API / Service Tests**
  - Input Validation
  - Authentication
  - Authorization
  - Session Management
  - Output Escaping (XSS)
  - Injection
  - Security Misconfiguration

Security on the Service Testing Layer

- Injection (SQL Injections)
- Broken Access Control (Authorization Layers)
- Misconfiguration - Cross Origin Resource Sharing (CORS)

DEVOXX United Kingdom

V4: Access Control (ASVS)

- **V4.1 General Access Control Design**
  - 4.1.1 Verify that the application enforces access control rules on a trusted service layer, …
  - 4.1.3 Verify that the principle of least privilege exists…
  - 4.1.5 Verify that access controls fail securely…

https://owasp.org/www-project-application-security-verification-standard

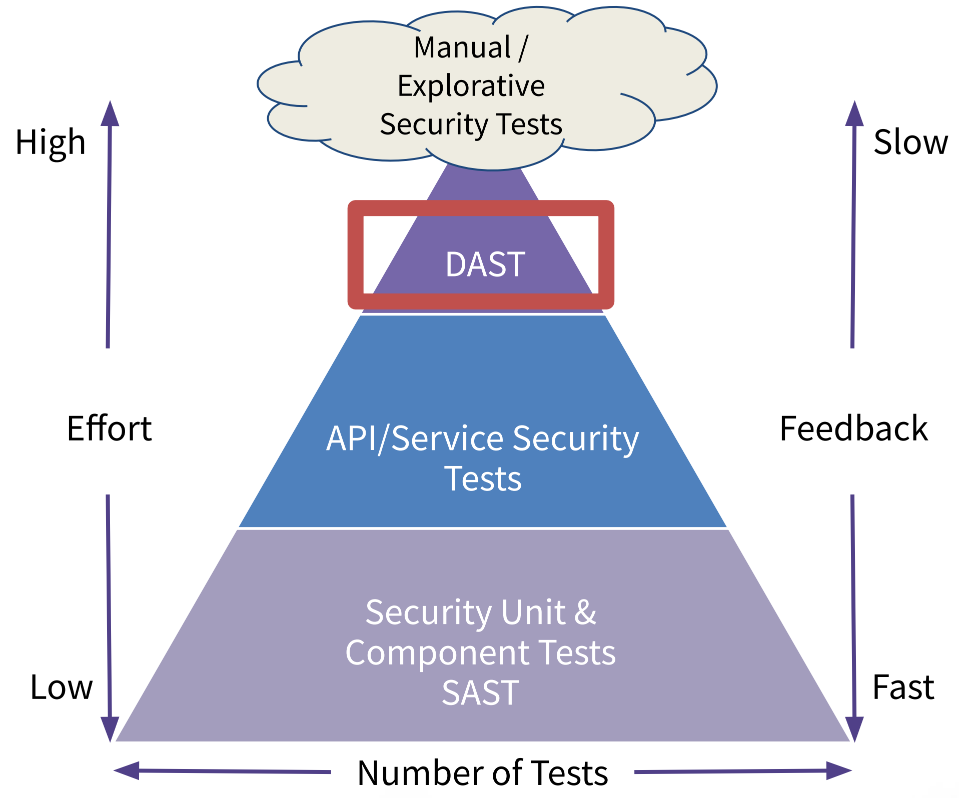
devoxx uk
the developer community conference

Demo

Service Test Layer

https://github.com/andifalk/bookmark-service

The Security-Testing Pyramid: DAST


Manual / Explorative Security Tests
DAST
API/Service Security Tests
Security Unit & Component Tests SAST
High
Effort
Low
Slow
Feedback
Fast
Number of Tests

- **Dynamic Application Security Testing (DAST)**
  - OWASP Zap / StackHawk
  - Portswigger Burp Suite
  - SQLMap
  - NMap
  - Gatling
  - …

Dynamic Application Security Testing

```
$ docker pull owasp/zap2docker-stable
```

- Baseline Scan
  - Runs the spider and passive scanning:
    zap-baseline.py -t https://www.example.com
- API Scan
  - Performs active scan against APIs defined by OpenAPI
    zap-api-scan.py -t https://example.com/openapi.json -f openapi
- Full Scan
  - Runs the ZAP spider and a full active scan (+ opt. ajax scan)
    zap-full-scan.py -t https://www.example.com

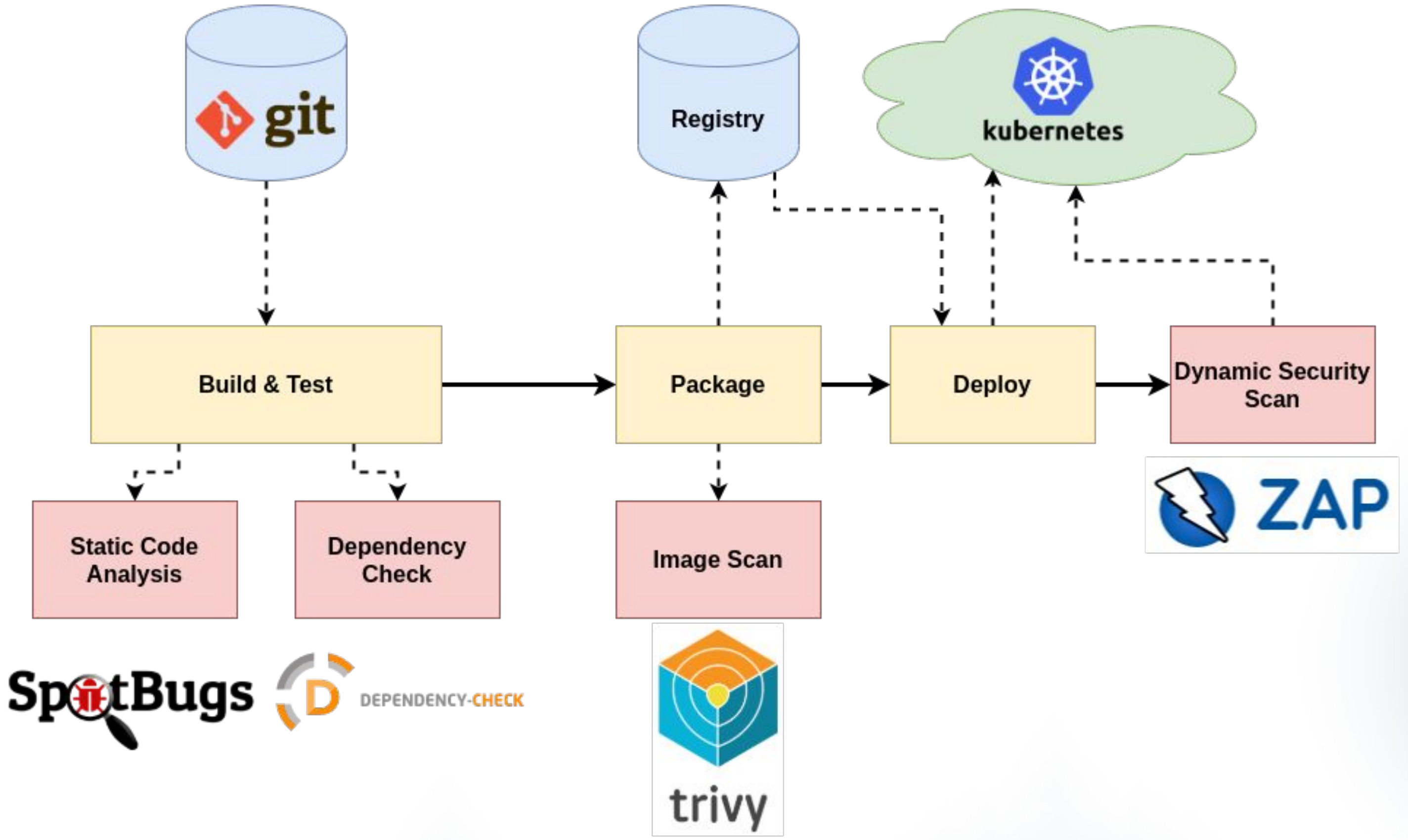
https://www.zaproxy.org/docs/docker

Demo

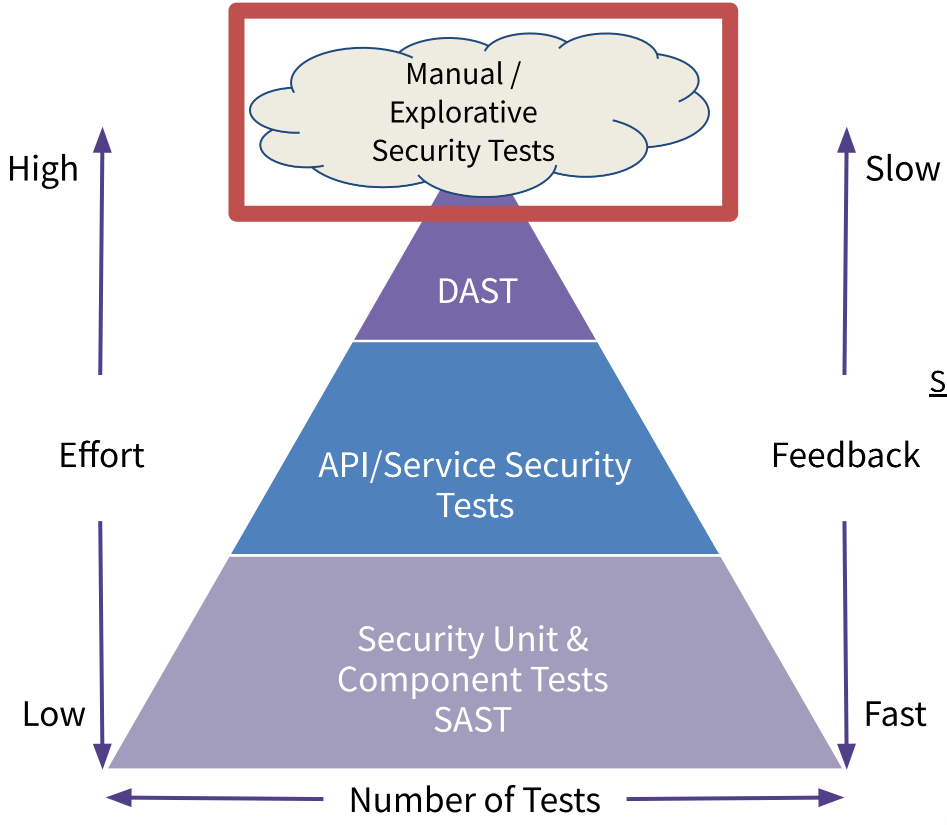
Dynamic Security Testing Layer

https://github.com/andifalk/bookmark-service

Automate all the “Security-Verification” Things!



The Security-Testing Pyramid: Explorative Security Tests

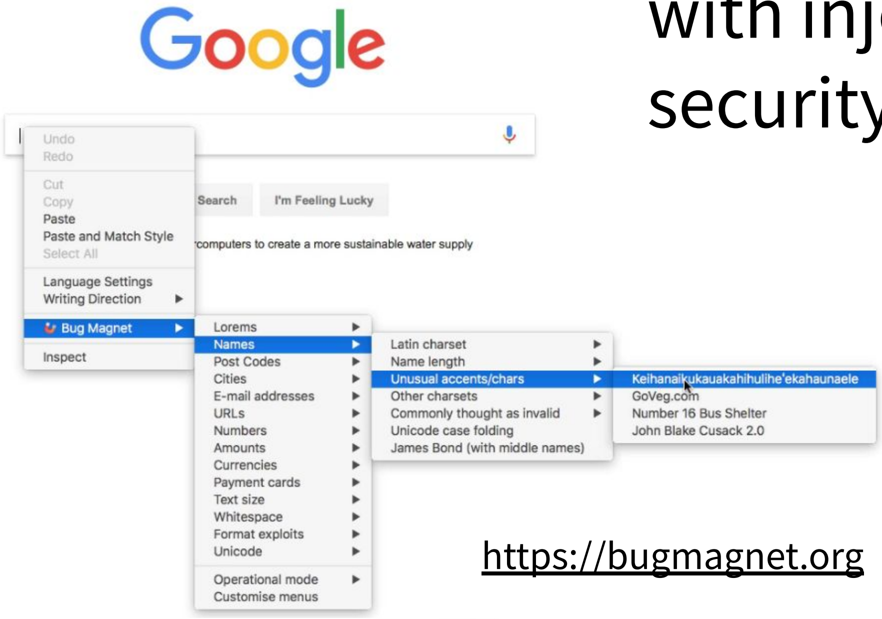

Manual / Explorative Security Tests
DAST
API/Service Security Tests
Security Unit & Component Tests SAST
High
Effort
Low
Slow
Feedback
Fast
Number of Tests

- Security Charters
- Security Code Reviews
- Pen-Tests

https://martinfowler.com/bliki/ExploratoryTesting.html
https://pragprog.com/titles/ehxta/explore-it
Secure-er Code Reviews with Seth & Ken! - OWASP DevSlop

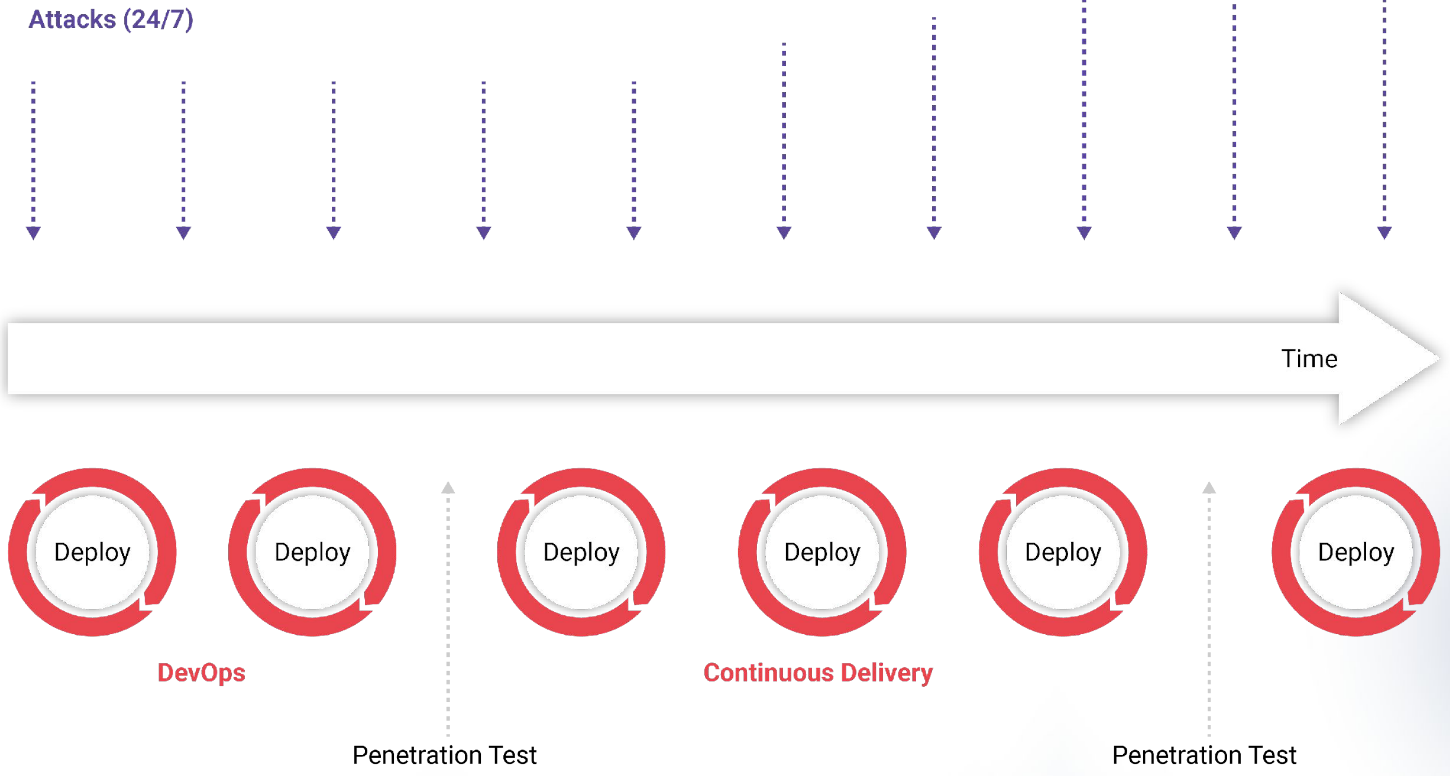
Explorative Security Tests

“Explore [*some functionality*] with injection attacks to discover security vulnerabilities”



https://bugmagnet.org

Penetration Tests



Summary

DEVOXX United Kingdom

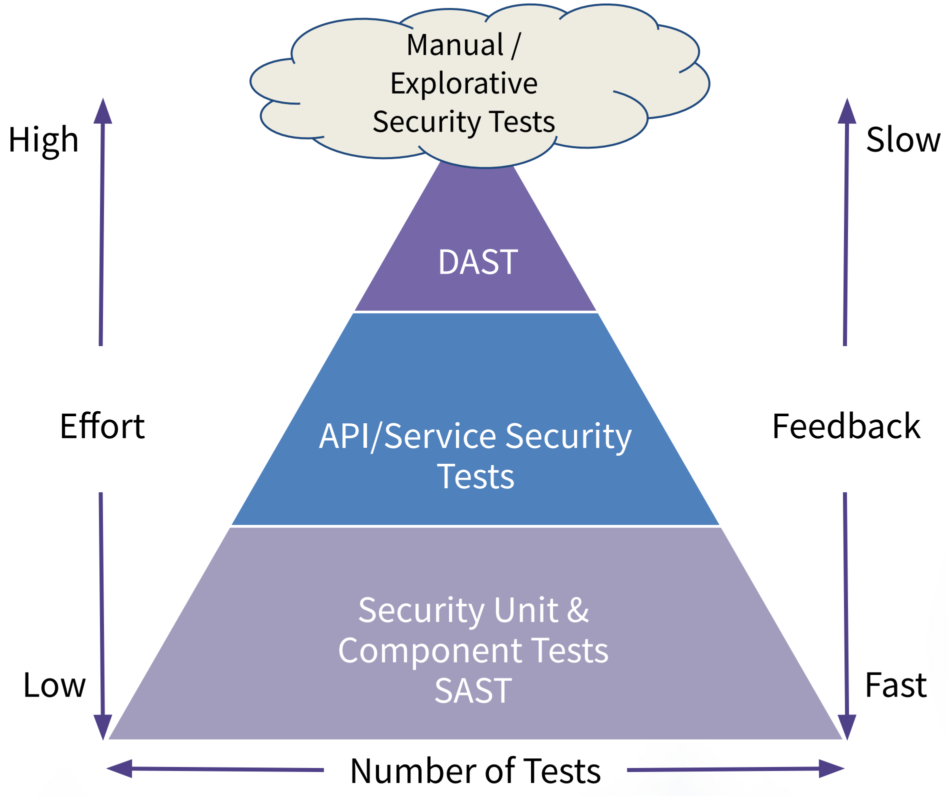
The Security-Testing Pyramid: Security On Every Layer

Security

Security

Security

Security


Manual / Explorative Security Tests
DAST
API/Service Security Tests
Security Unit & Component Tests SAST
High
Effort
Low
Slow
Feedback
Fast
Number of Tests

devoxx uk
the developer community conference

DevOps - The “Sec” Is Silent

Devs
Security

Please Show Some <u>Empathy</u>!

DEVOXX United Kingdom

Get all testing code samples:

https://github.com/andifalk/bookmark-service

Scan the code for my contact details

Q & A

devoxx uk
the developer community conference