

Event

Security Meetup:

Catching Vulnerabilities & AI-Driven DevSecOps



Google Developer Group
Stuttgart

22 January 2026





CGI
22.01.2026

66  Google Developer Group
Stuttgart

Catch What E2E Tests Miss:
Real-World Security Testing
with ZAP

Marcel Haag
CGI Deutschland

99

Agenda

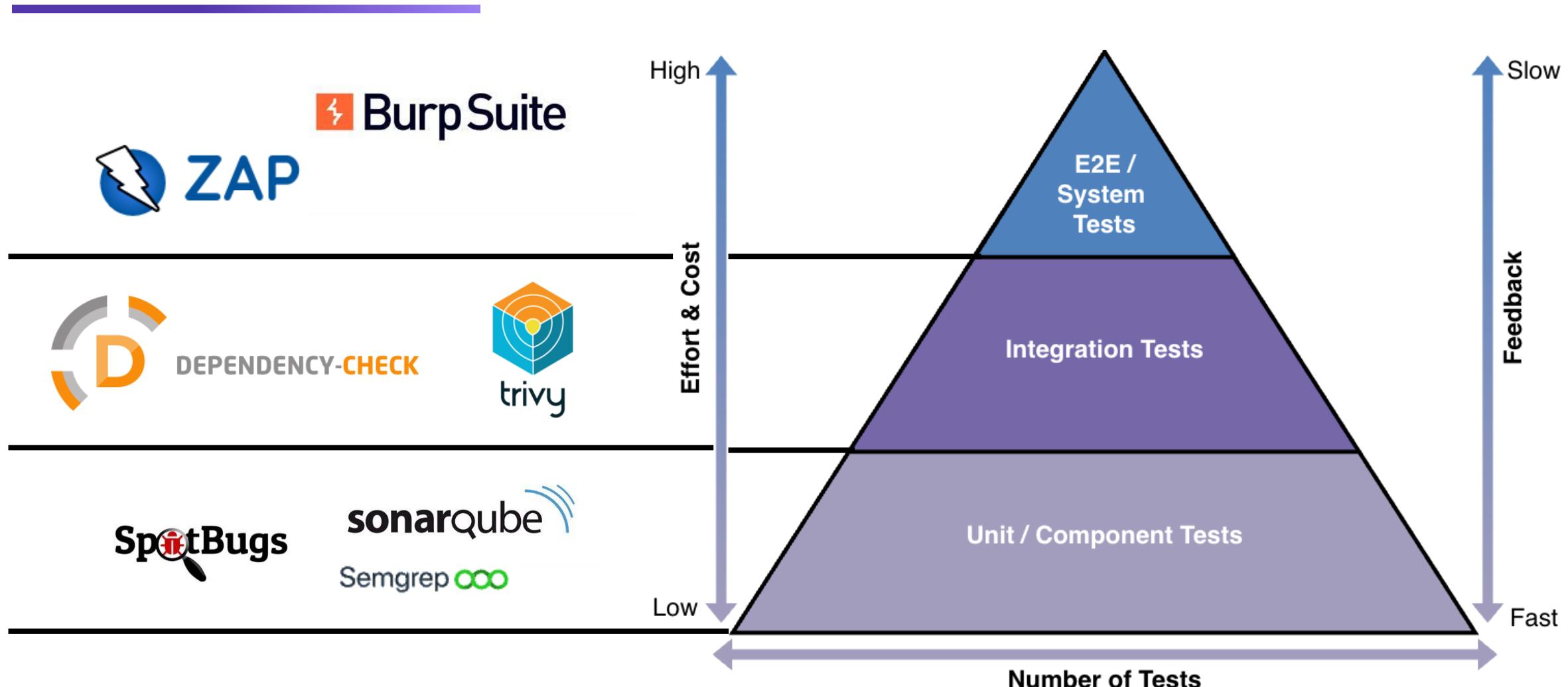
- 1 Introduction: Testing Pyramid & DevSecOps Tooling
- 2 Why E2E Tests are not enough
- 3 Introducing DAST in the SDLC
- 4 ZAP - *One Proxy to attack them all*
- 5 CI/CD Integration with GitHub Actions + **Demo**
- 6 Bonus: Pipeline Hardening



Introduction Testing Pyramid & DevSecOps Tooling

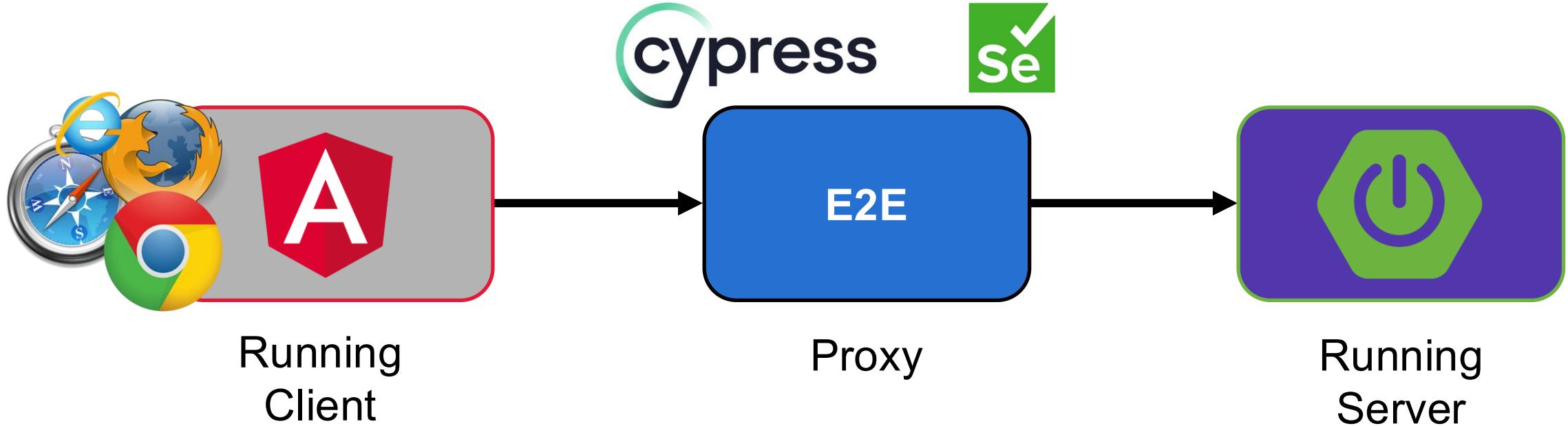
DevSecOps Tooling

Testing Pyramid



Why E2E Tests are not enough

Dynamic Application Security Testing (DAST)



E2E Tests

Does

- ✓ validate real user journeys
- ✓ verify system integration
- ✓ test expected behaviour
- ✓ use known inputs

Can a user sign up / login,
and complete a purchase?

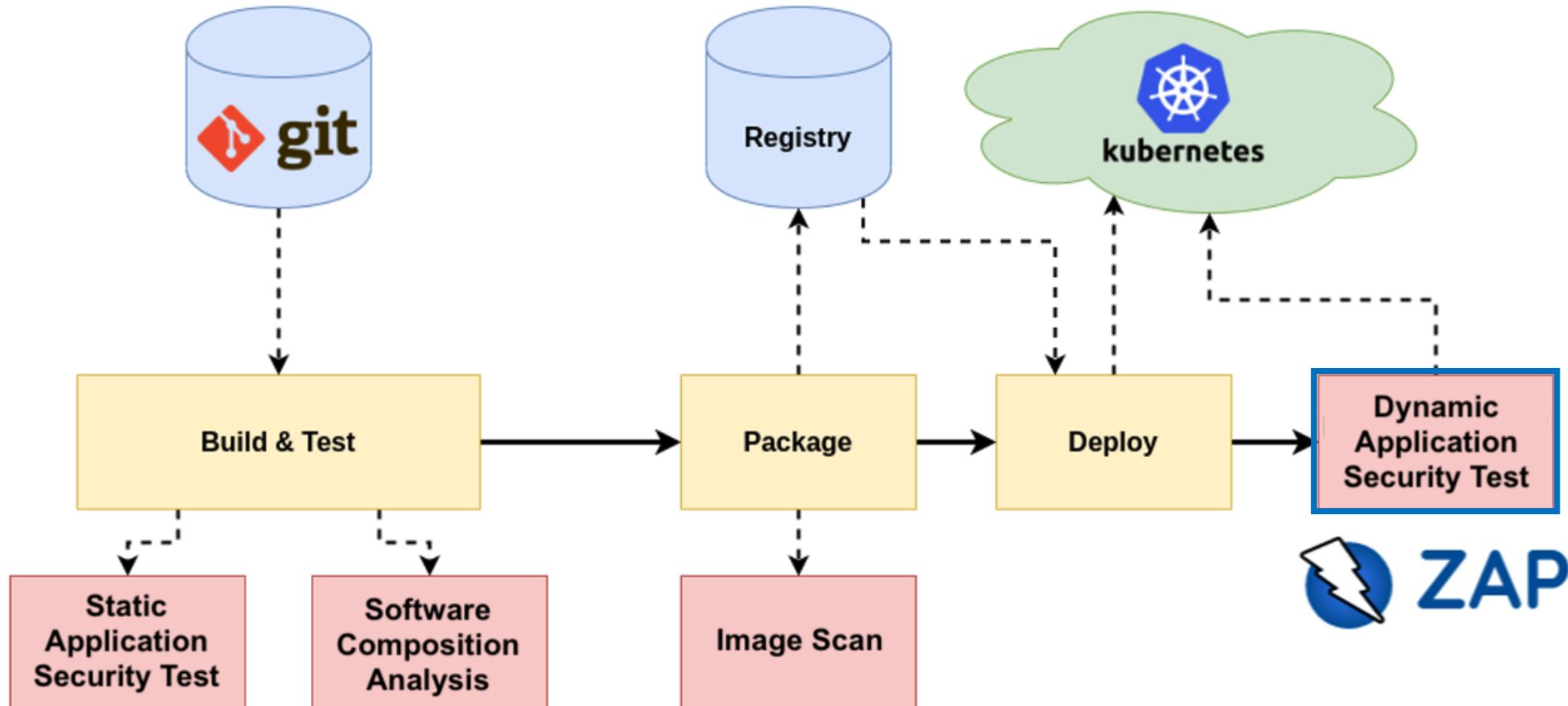
Doesn't do

- fuzzing of inputs
- enumerate endpoints
- abuse auth logic
- chain attacks

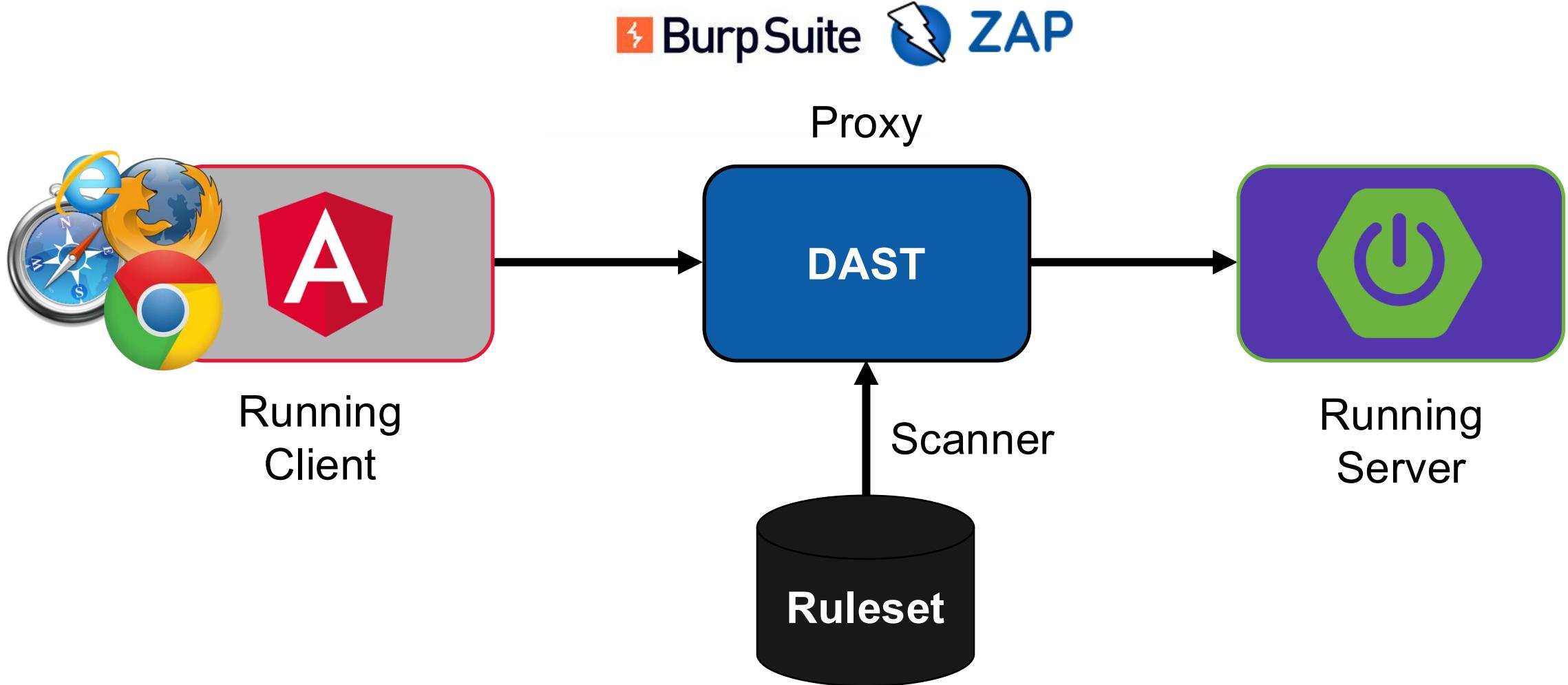


Introducing DAST in the SDLC

ZAP in CI-/CD-Pipeline



Dynamic Application Security Testing (DAST)



ZAP

- One Proxy to attack them all

Zed Attack Proxy (ZAP)

The screenshot shows the OWASP ZAP 2.12.0 interface. The main window title is "Untitled Session - OWASP ZAP 2.12.0". The left sidebar shows "Contexts" and "Sites" for "Default Context" and "http://192.168.0.8:3000". The "Sites" section lists various URLs and their status codes (e.g., GET:/, GET:103.js, GET:api, GET:assets, GET:favicon.ico, GET:font-mfizz.woff, GET:ftp, GET:home, GET:main.js, GET:Materialicons-Regular.woff2, GET:polyfills.js, GET:profile). The central panel is titled "Automated Scan" with a sub-instruction: "This screen allows you to launch an automated scan against an application - just enter its URL below and press 'Attack'. Please be aware that you should only attack applications that you have been specifically been given permission to test." It includes fields for "URL to attack" (set to "http://192.168.0.8:3000/"), "Use traditional spider" (checked), "Use ajax spider" (unchecked), and "Attack" (button). The bottom of the central panel shows "Progress: Not started". The bottom navigation bar includes tabs for History, Search, Alerts, Output, WebSockets, Fuzzer, Spider, and AJAX Spider, along with a "Filter: OFF" button and an "Export" link. The "History" tab is active, displaying a table of requests:

ID	Source	Req. Timestamp	Method	URL	Code	Reason	RTT	Size Resp. Body	Highest Alert	Note	Tags
7.905	Auth	11/03/2023, 20:18:41	GET	http://192.168.0.8:3000/profile	500	Internal Server Error	6 ms	1.279 bytes			Authentication
7.915	Auth	11/03/2023, 20:18:45	GET	http://192.168.0.8:3000/profile	500	Internal Server Error	5 ms	1.279 bytes			Authentication
7.917	Auth	11/03/2023, 20:18:46	GET	http://192.168.0.8:3000/profile	500	Internal Server Error	5 ms	1.279 bytes			Authentication
7.918	Auth	11/03/2023, 20:18:46	GET	http://192.168.0.8:3000/profile	500	Internal Server Error	4 ms	1.279 bytes			Authentication
7.930	Auth	11/03/2023, 20:18:50	GET	http://192.168.0.8:3000/profile	500	Internal Server Error	4 ms	1.279 bytes			Authentication
7.931	Auth	11/03/2023, 20:18:51	GET	http://192.168.0.8:3000/profile	500	Internal Server Error	5 ms	1.279 bytes			Authentication
7.933	Auth	11/03/2023, 20:18:51	GET	http://192.168.0.8:3000/profile	500	Internal Server Error	7 ms	1.279 bytes			Authentication
7.938	Auth	11/03/2023, 20:18:52	GET	http://192.168.0.8:3000/profile	500	Internal Server Error	17 ms	1.279 bytes			Authentication
7.948	Auth	11/03/2023, 20:18:56	GET	http://192.168.0.8:3000/profile	500	Internal Server Error	16 ms	1.279 bytes			Authentication
7.949	Auth	11/03/2023, 20:18:56	GET	http://192.168.0.8:3000/profile	500	Internal Server Error	5 ms	1.279 bytes			Authentication
7.956	Proxy	11/03/2023, 20:18:34	GET	http://192.168.0.8:3000/socket.io/?EIO=4&transport=polling	200	OK	24,87 s	1 bytes	Medium		
7.957	Proxy	11/03/2023, 20:18:59	POST	http://192.168.0.8:3000/socket.io/?EIO=4&transport=polling	200	OK	1 ms	2 bytes	Medium		
7.963	Proxy	11/03/2023, 20:18:59	GET	http://192.168.0.8:3000/socket.io/?EIO=4&transport=polling	200	OK	25 s	1 bytes	Medium		
7.964	Proxy	11/03/2023, 20:19:24	POST	http://192.168.0.8:3000/socket.io/?EIO=4&transport=polling	200	OK	2 ms	2 bytes	Medium		
7.965	Proxy	11/03/2023, 20:19:24	GET	http://192.168.0.8:3000/socket.io/?EIO=4&transport=polling	504	Gateway Timeout	20,01 s	247 bytes			
7.966	Proxy	11/03/2023, 20:19:44	POST	http://192.168.0.8:3000/socket.io/?EIO=4&transport=polling	400	Bad Request	13 ms	41 bytes	Medium		JSON
7.967	Proxy	11/03/2023, 20:19:45	GET	http://192.168.0.8:3000/socket.io/?EIO=4&transport=polling	200	OK	17 ms	96 bytes	Low		
7.968	Proxy	11/03/2023, 20:19:45	POST	http://192.168.0.8:3000/socket.io/?EIO=4&transport=polling	200	OK	2 ms	2 bytes	Medium		
7.969	Proxy	11/03/2023, 20:19:45	GET	http://192.168.0.8:3000/socket.io/?EIO=4&transport=polling	200	OK	0 ms	32 bytes	Medium		
7.970	Proxy	11/03/2023, 20:19:45	GET	http://192.168.0.8:3000/socket.io/?EIO=4&transport=websocket	101	Switching Protocols	1 ms	0 bytes			

At the bottom, there are "Alerts" (8 alerts, 5 medium, 3 low) and "Main Proxy: localhost:8090" status indicators.

Passive & Active Scans
Script Integration
Automation
Reporting
Spidering
Fuzzing

Passive Scan

Focus

- Analyzes the given page/domain
- Identifies potential vulnerabilities
- Quick scan for:
 - Vulnerable HTML tags (Sinks)
 - Missing headers
 - Misconfiguration (Cookie, CSP)



Active Scan

Focus

- Uses crawlers and web fuzzers
- Verifies identified vulnerabilities
- JS analysis for further vulnerabilities
- Fuzzes request bodies and parameters for:
 - Various injection attacks
 - XSS, SQLi, XXE, ...
 - Other common web vulnerabilities.



Report Generation

Contains

- Summary of all findings
- Finding details
 - Description
 - URL metadata
 - (HTTP method, param, etc.)
 - References (rfc)
 - CWE ID
 - Plugin ID

<https://www.zaproxy.org/docs/desktop/addons/report-generation/>



Site: <https://saftladen.automatisier.bar>

Generated on Tue, 25 Nov 2025 15:53:13

ZAP Version: D-2025-11-24

ZAP by [Checkmarx](#)

Summary of Alerts

Risk Level	Number of Alerts
High	1
Medium	3
Low	4
Informational	5
False Positives:	0

Summary of Sequences

For each step: result (Pass/Fail) - risk (of highest alert(s) for the step, if any).

Alerts

Name	Risk Level	Number of Instances
SQL Injection	High	2
Content Security Policy (CSP) Header Not Set	Medium	1
Cross-Domain Misconfiguration	Medium	Systemic
Integer Overflow Error	Medium	1
A Server Error response code was returned by the server	Low	16

CI/CD Integration with GitHub Actions

ZAP Baseline Scan

- Focuses on identifying potential vulnerabilities
- Designed to detect known security issues:
 - XSS Protection (Header)
 - SQL Injections
 - Path Traversal
 - HTTP Security Headers
 - Unvalidated Redirects and Forwards
 - Insecure Deserialization
 - ...

(Passive Scan)

```
zap_baseline_scan:
  name: "ZAP Baseline Job"

  runs-on: ubuntu-latest

  steps:
    - name: "ZAP Baseline Scan"
      uses: zaproxy/action-baseline@v0.9.0
      with:
        token: ${{ secrets.GITHUB_TOKEN }}
        docker_name: 'ghcr.io/zaproxy/zaproxy:stable'
        target: 'https://example.com'
        rules_file_name: '.zap/rules.tsv'
        artifact_name: zap_baseline_scan
        cmd_options: '-a'
```

<https://www.zaproxy.org/docs/docker/baseline-scan/>

ZAP API & Full Scan

- Focuses on scanning RESTful APIs
- Supported formats:
 - Openapi (DEFAULT)
 - Soap
 - GraphQL
- Searches for a variety of vulnerabilities:
 - SQL Injections
 - Authentication issues
 - Insecure direct object references (IDOR)
 - Broken Access Control
 - Sensitive Operations Without Confirmation
 - ...

(Active Scans)

```
zap_api_scan:  
  name: "ZAP API Job"  
  
  runs-on: ubuntu-latest  
  
  steps:  
  
    - name: "ZAP API Scan"  
      uses: zaproxy/action-api-scan@v0.5.0  
      with:  
        token: ${ secrets.GITHUB_TOKEN }  
        docker_name: 'ghcr.io/zaproxy/zaproxy:stable'  
        format: openapi  
        target: 'https://example.com'  
        rules_file_name: '.zap/rules.tsv'  
        artifact_name: zap_api_scan  
        cmd_options: '-a'
```

<https://www.zaproxy.org/docs/docker/api-scan/>

<https://www.zaproxy.org/docs/docker/full-scan/>

With Docker

Different scans types:

- zap-full-scan.py
- zap-baseline.py
- zap-api-scan.py

Options:

- t target domains / APIs
- f file format
- r report file
- g config for the context
- z set options (e.g. Auth header)

<https://www.zaproxy.org/docs/docker/about/>
<https://deepwiki.com/zaproxy/zaproxy/5.1-scan-rules>

© 2026 CGI Inc.

```
zap_docker_api_scan:  
  name: "ZAP API Job (Dockerized)"  
  
  runs-on: [ xentry, medium ]  
  
  steps:  
    - uses: actions/checkout@v2  
    - name: "Pull ZAP Image"  
      run:  
        docker pull zaproxy/zap-weekly  
  
    - name: "Run ZAP API Scan (json)"  
      run: |  
        set +e  
        docker run -v $(pwd):/zap/wrk/:rw -t zaproxy/zap-weekly \  
          zap-api-scan.py -t OpenAPI.json -f openapi \  
          -r zap_report.html -g api-active-scan.conf -z options.prop  
        echo "ZAP completed with exit code $?"  
        set -e  
  
    - name: Upload ZAP Report  
      uses: actions/upload-artifact@v3  
      with:  
        name: zap-report  
        path: zap_report.html
```

Demo

[https://github.com/
marcel-haag/dev-
sec-ops-demos](https://github.com/marcel-haag/dev-sec-ops-demos)



Bonus: Pipeline Hardening

ZAP Baseline Job summary				
🛡 Network Activity Monitored by StepSecurity Harden-Runner				
Network calls made by the runner during this job run. These were automatically monitored and logged in real-time by StepSecurity Harden-Runner .				
Process	Destination	Port	Status	Timestamp
java	saftladen.automatisier.bar	443	<input checked="" type="checkbox"/> Allowed	Nov 27 2025 11:27:15
java	raw.githubusercontent.com	443	<input checked="" type="checkbox"/> Allowed	Nov 27 2025 11:27:14
java	news.zaproxy.org	443	<input checked="" type="checkbox"/> Allowed	Nov 27 2025 11:27:13
java	cfu.zaproxy.org	443	<input checked="" type="checkbox"/> Allowed	Nov 27 2025 11:27:03
dockerd	production.cloudflare.docker.com	443	<input checked="" type="checkbox"/> Allowed	Nov 27 2025 11:26:32



Pipeline Hardening

Recommended Configuration to Harden the Runner

- Block egress traffic: Only allow calls to allowed endpoints

Initial baseline created. It will be updated as more job runs are monitored.

```
- name: Harden Runner
  uses: step-security/harden-
runner@95d9a5deda9de15063e7595e9719c11c38c90ae2 # v2.13.2
  with:
    egress-policy: block
    allowed-endpoints: >
      auth.docker.io:443
      cfu.zaproxy.org:443
      github.com:443
      news.zaproxy.org:443
      production.cloudflare.docker.com:443
      raw.githubusercontent.com:443
      registry-1.docker.io:443
      saftladen.automatisier.bar:443
      tel.zaproxy.org:443
```

Thank you
for listening!

Questions?



Marcel Haag

marcel.haag@cgi.com

