9 December 2021

Automate API Protection with Security as Code

Colin Domoney

API Security Research Specialist & Developer Advocate

@colindomoney

Introduction

About the Speaker

Colin Domoney

API Security Research Specialist & Developer Advocate

Editor of APISecurity.io
Cyberbroof, Veracode, CA, Deutsche Bank

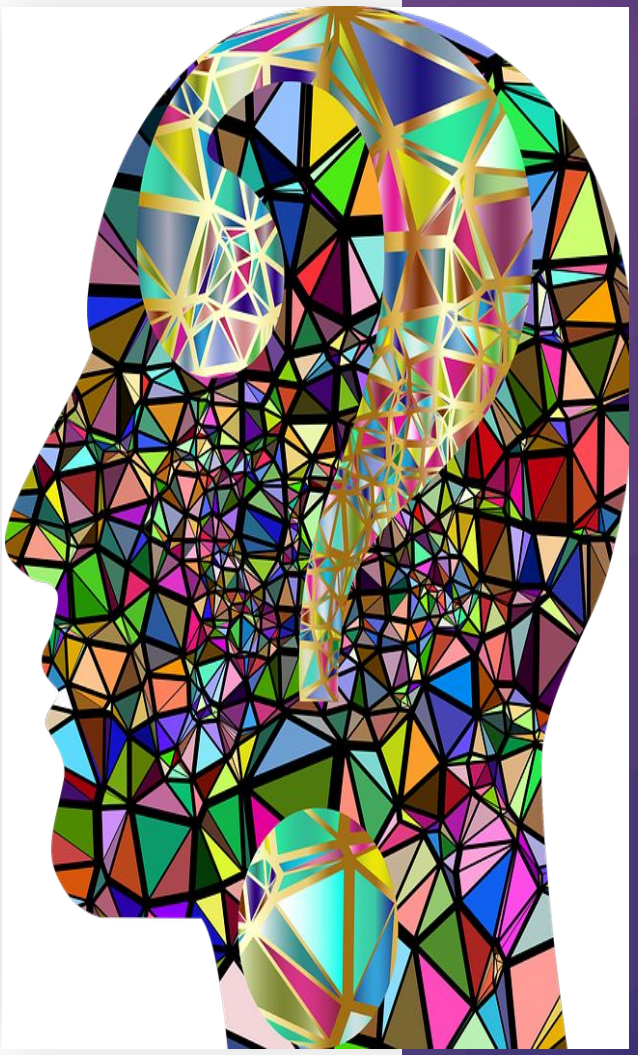
Housekeeping Rules

- All attendees muted
- Questions via chat
- Recording will be shared
- Polling questions

Question One:

How many internal and external APIs have you deployed in your organization?

1. Less than 50
2. 500 to 500
3. 500 plus
4. Don’t know



Q 1: How many internal and external APIs have you deployed in your organization?

Agenda

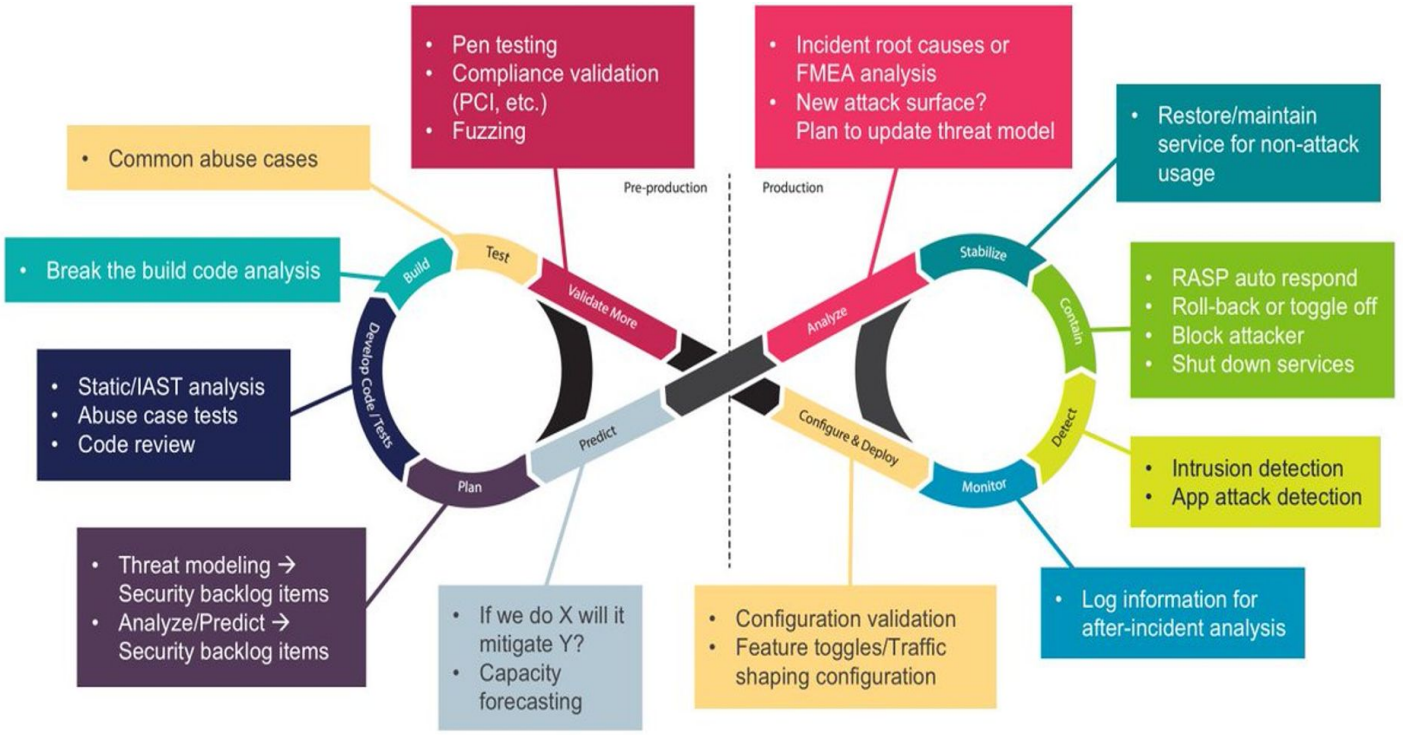
- Challenges to scaling security
  - The promise of DevSecOps
  - Where is the security team located – **Sec**DevOps, DevOps**Sec**, Dev**Sec**Ops ?
- Everything-as-Code
  - Infrastructure-as-Code
  - GitOps
  - Security-as-Code for APIs
- 42Crunch protections
  - Introduction to firewall
  - Security protections
  - OAS demonstration
- Live demo
- Use cases / Benefits

Challenges to scaling software security

What is DevSecOps ?

My definition of DevSecOps:

- Requires **tight integration** into DevOps process
- Needs to be **continuous and automated**
- Needs to be **low latency**
- Traditional siloed Security is no longer suitable
- Security **enables delivery**, not act as an impediment
- Security is **omnipresent** and **everyone's responsibility**
- Challenges for People, Process and Technology

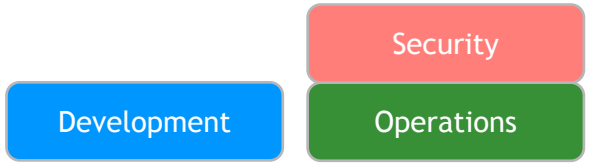


DevSecOps team topologies

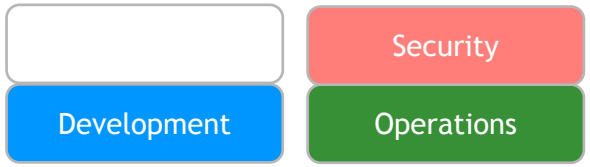
No security at all !

Security gets in the way

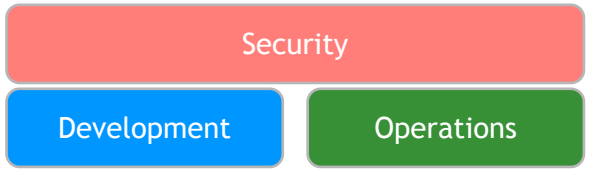
Security happens too late



Developers co-opted into security



Nirvana! DevSecOps



API security is a growing challenge

API Security Newsletter Archive

18 November, 21
Issue 160: Vulnerability in AWS API gateway, Kubernetes API access hardening guide

10 November, 21
Issue 159: Vulnerability in GoCD CI/CD platform, views on full lifecycle API security, articles on API security and sprawl

3 November, 21
Issue 158: Data of 400 000 students exposed, 1 million sites affected by plugin vulnerabilities, views on GraphQL

27 October, 21
Issue 157: Unsafe defaults in Prometheus, mapping API attack surfaces, OpenAPI file trend analysis

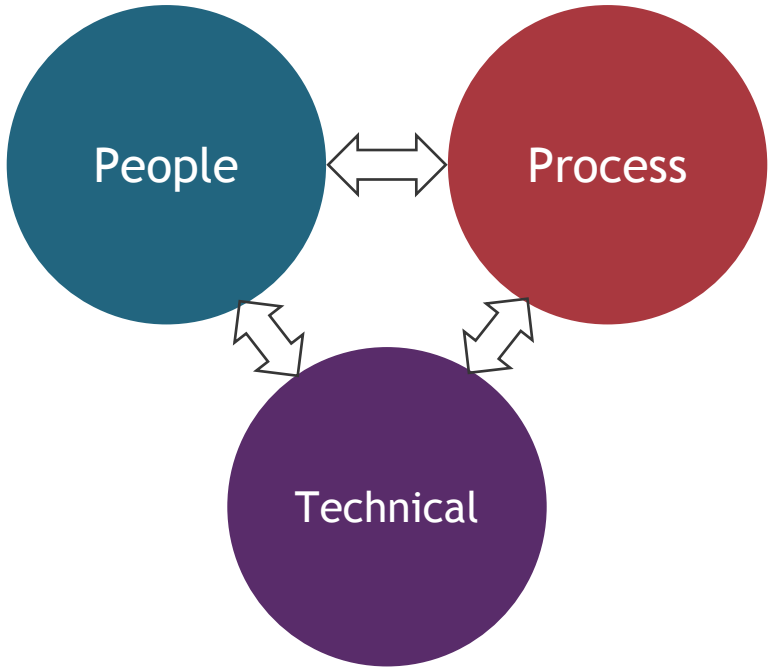
20 October, 21
Issue 156: FHIR APIs vulnerable to abuse, 3D printers facing hijacking risk, API security webinar

13 October, 21
Issue 155: Vulnerability in BrewDog mobile app, APIClarity at KubeCon, API attacks in Open Banking

https://apisecurity.io/

Why APIs are insecure

- Lack of skilled API developers
- Lack of awareness of attack methods and threats
- Shortage of security professionals



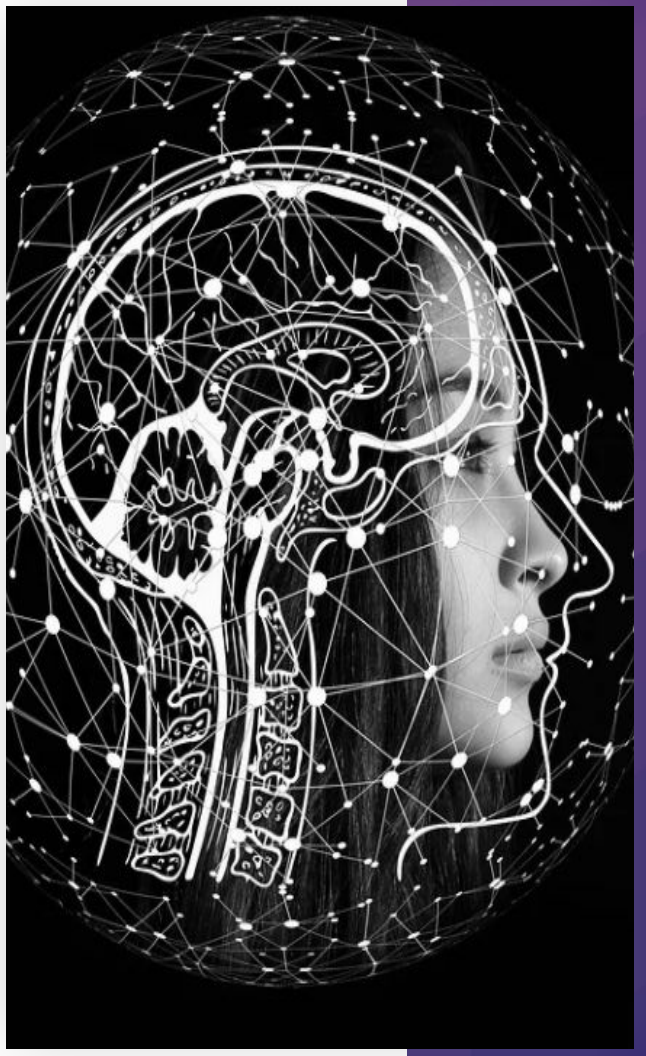
- DevSecOps not sufficiently developed or matured
- Security seen as an afterthought

- Existing security tools not suited to APIs
- APIs deployed too rapidly to test fully
- Complex frameworks inadvertently expose risk

Question Two:

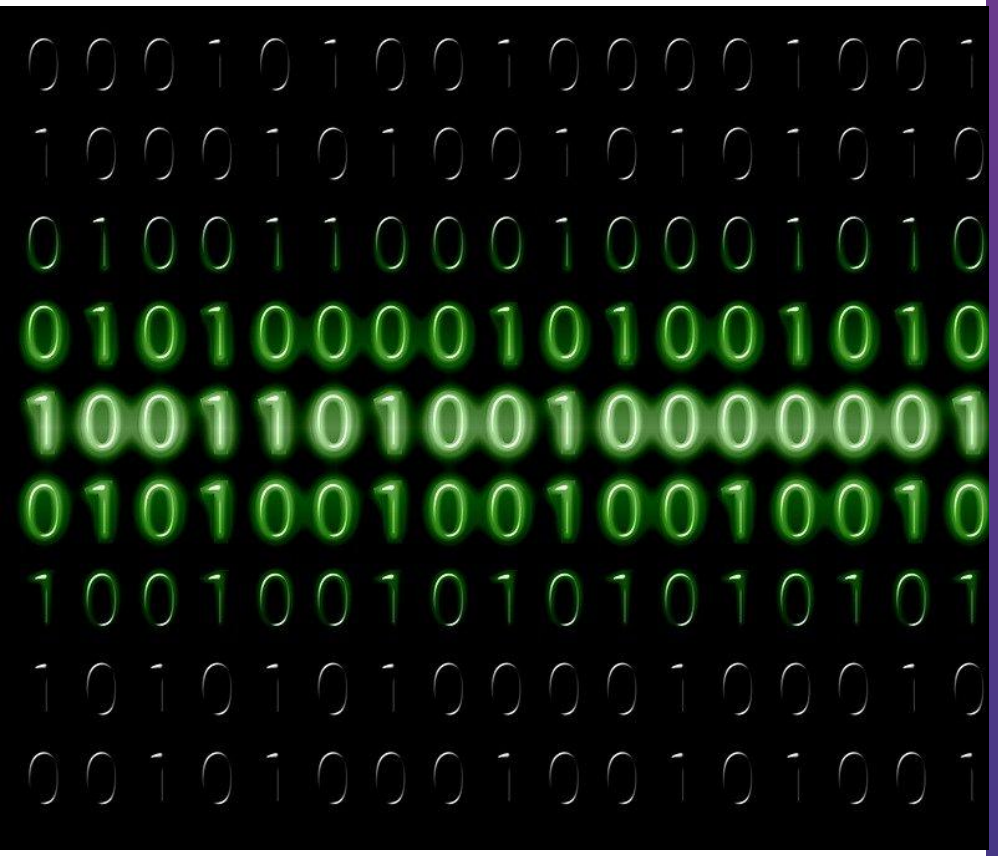
In your organization who is primarily responsible for API security?

1. Development teams
2. Security teams
3. API team
4. External consultants/providers
5. No-one in particular



Q 2: In your organization who is PRIMARILY responsible for API security?

- Development Teams — 44%
- Security Teams — 24.5%
- API Team — 18%
- External consultants / providers — 3.5%
- No one in particular — 10%

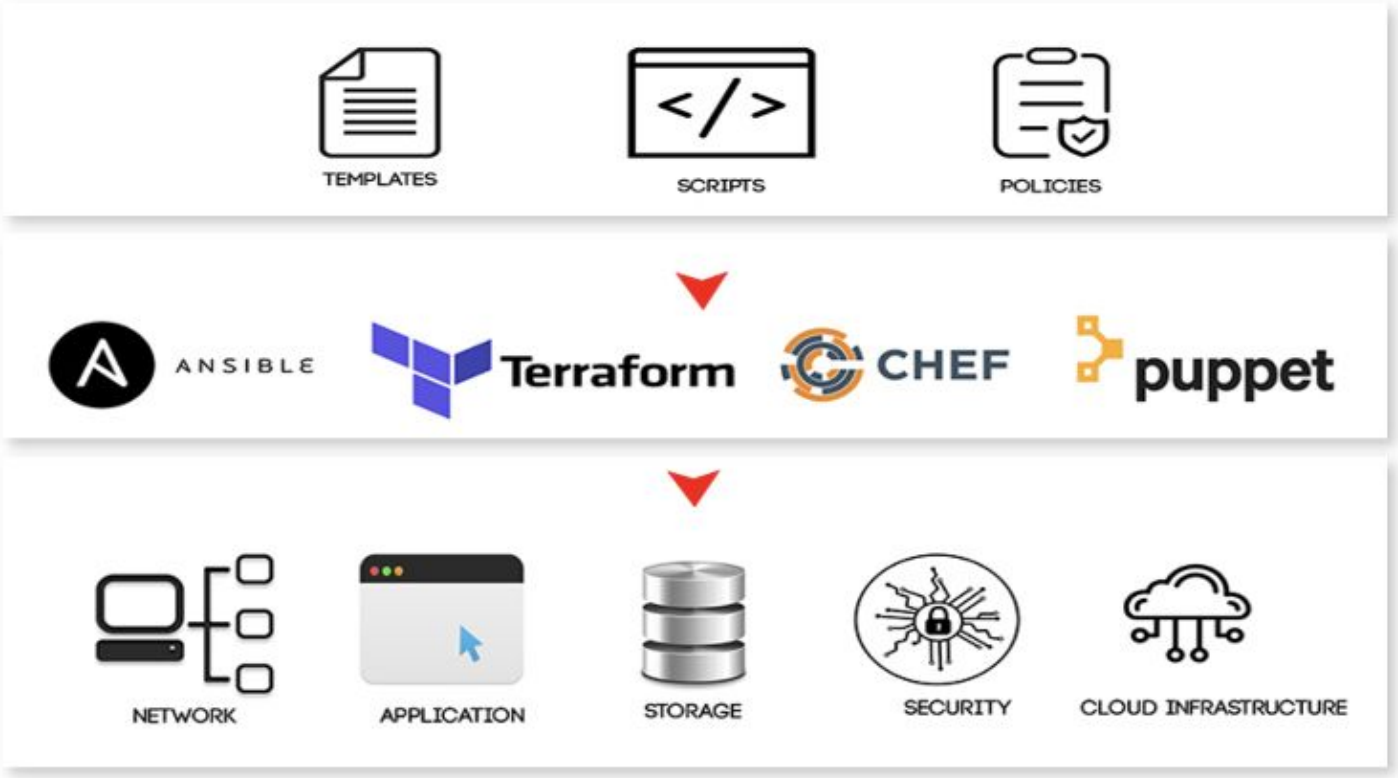


Everything-as-Code

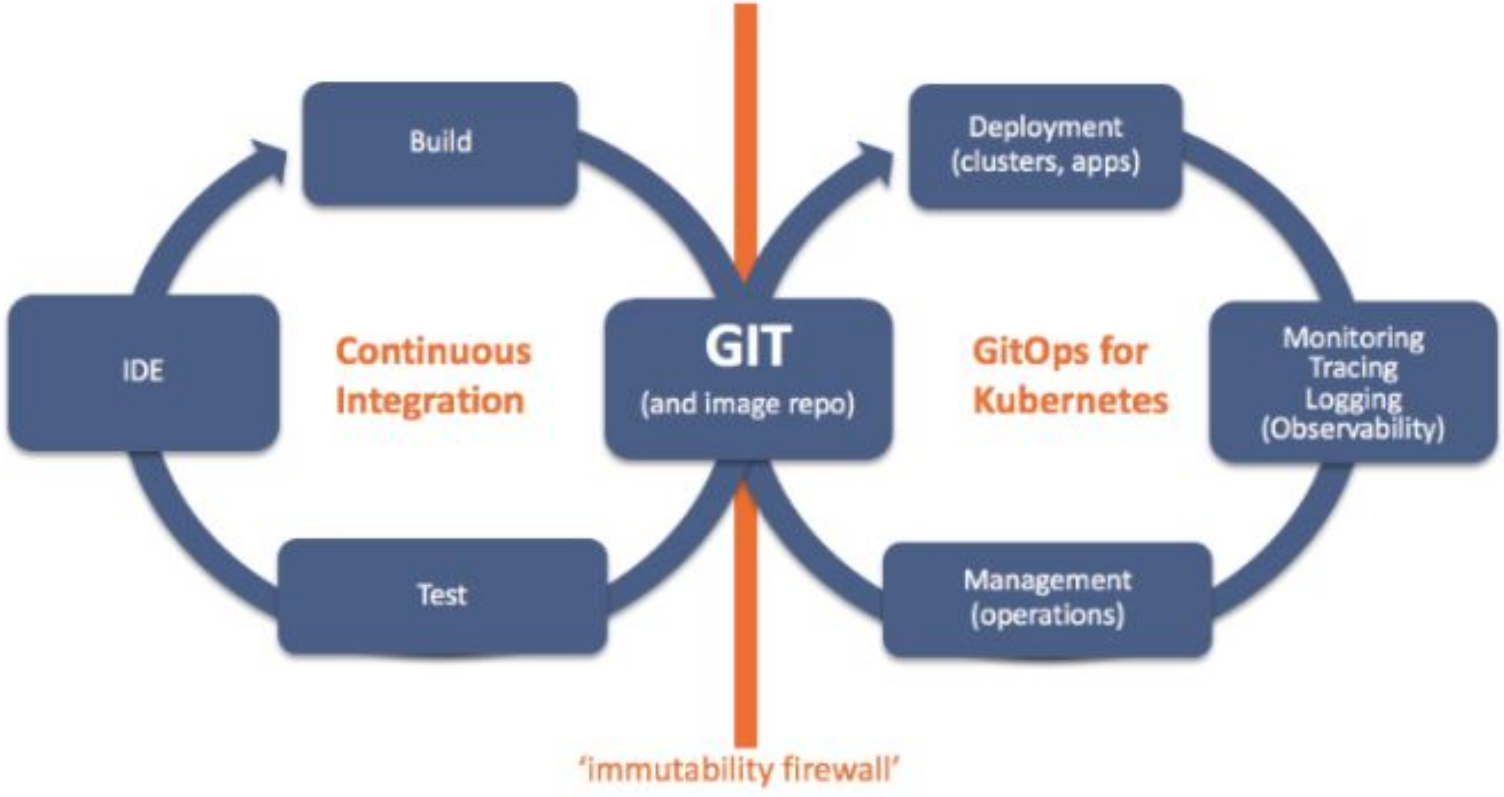
Case study #1: Infrastructure-as-Code

Benefits of Infrastructure-as-Code

- Cost reduction
- Speed
- Reduced risk
- Test
- Stable and scalable environments
- Accountability
- Configuration consistency
- Documentation
- Enhanced security



Case study #2: GitOps



Git as the single source of truth of a system's desired state

GitOps Diffs compare desired state with observed state (eg Kubediff, Terradiff, Canary..)

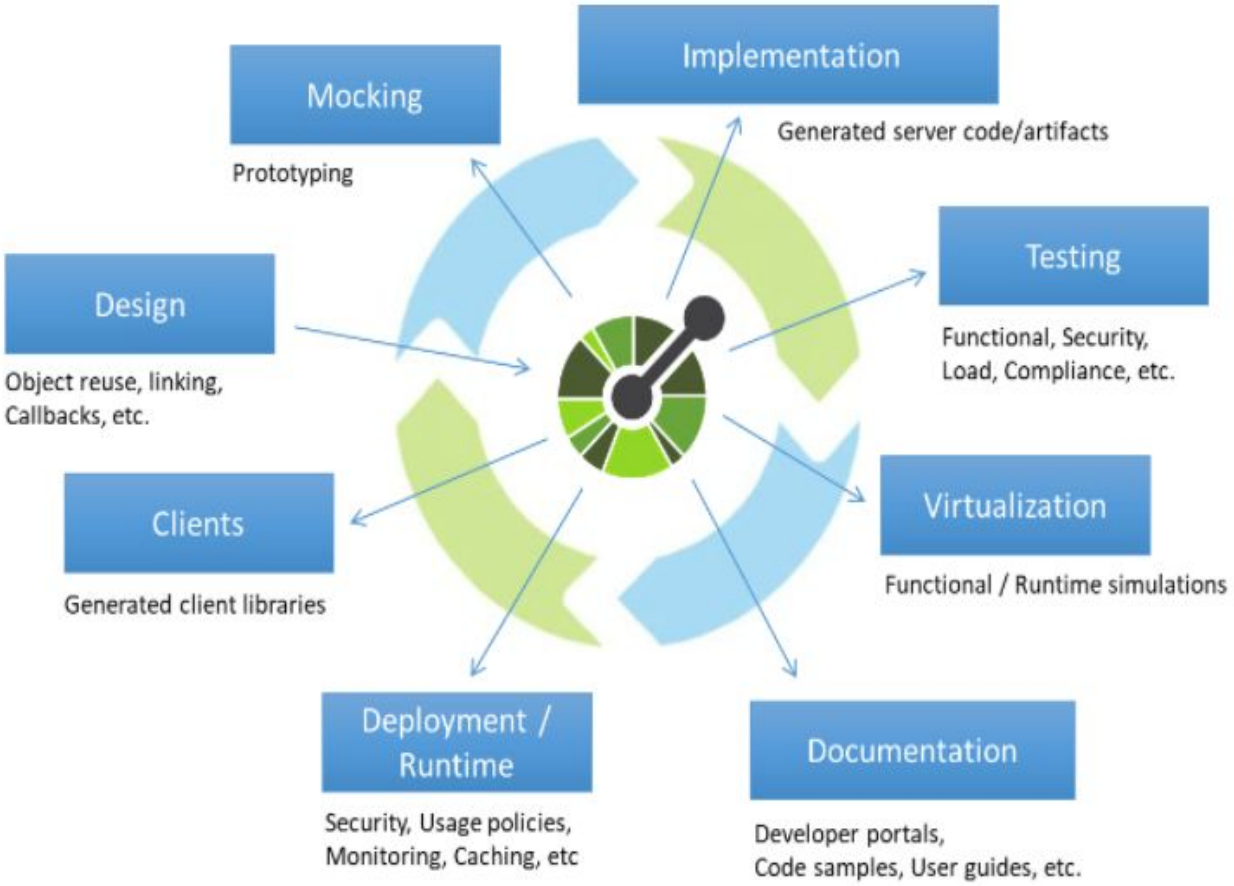
ALL intended operations are committed by pull request, for all environments

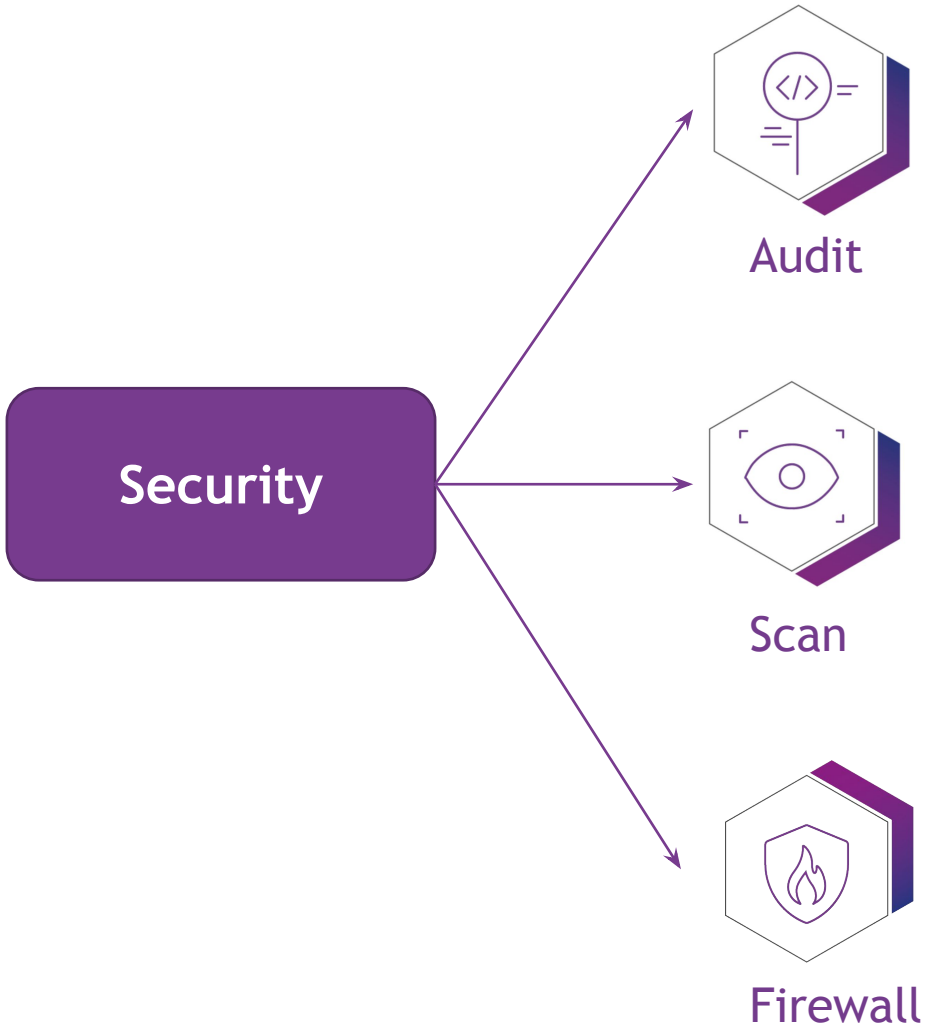
ALL diffs between GIT and observed state lead to (auto) convergence using tools like K8s

ALL changes are observable, verifiable and audited indisputably, with rollback & D/R

https://www.weave.works/blog/what-is-gitops-really

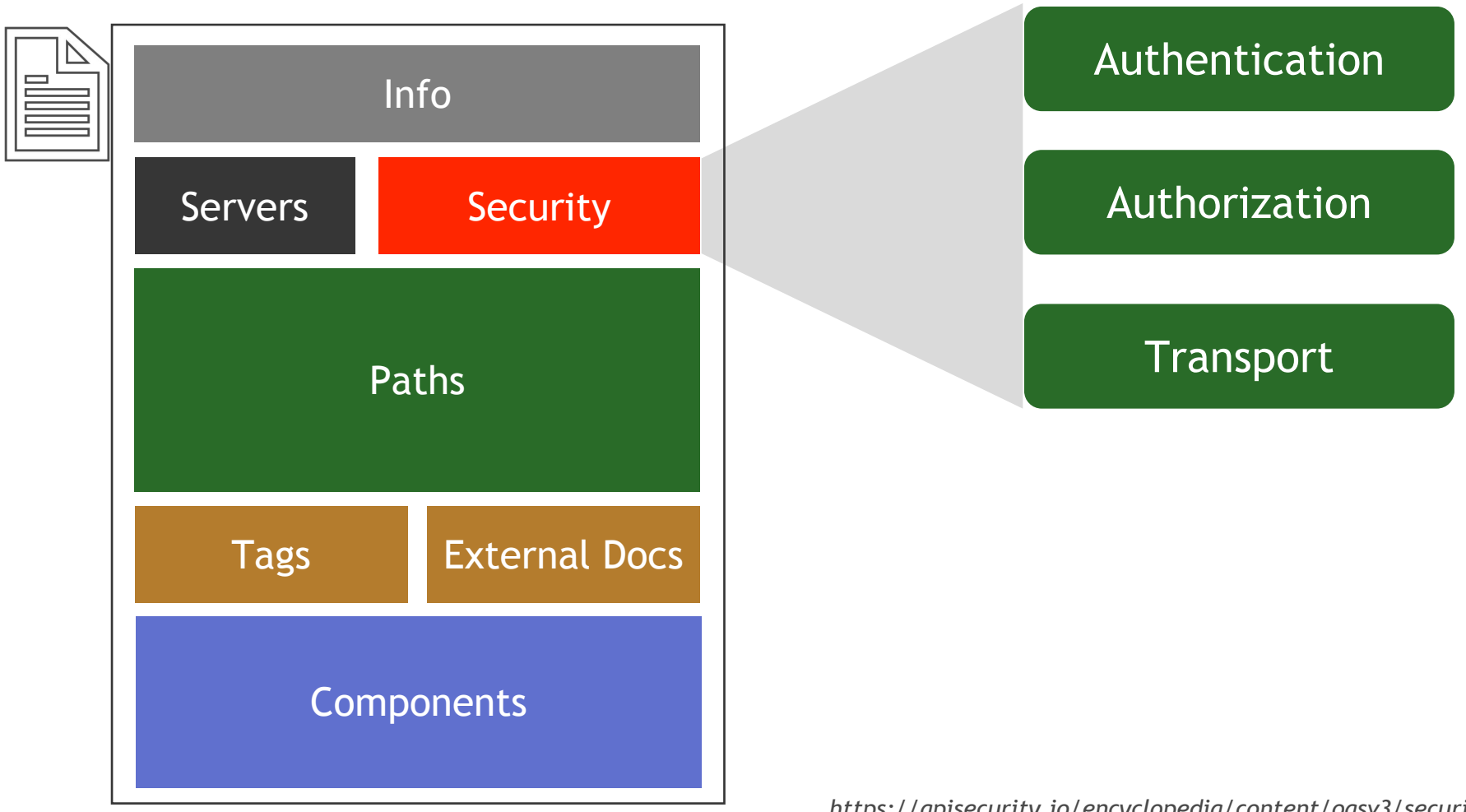
The API advantage – the API specification





https://swagger.io/blog/api-strategy/benefits-of-openapi-api-development/

Security in the OAS



https://apisecurity.io/encyclopedia/content/oasv3/security/security

42Crunch protections

#1: Automatic contract enforcement

- x-42c-deactivate-allowlist
- x-42c-request-allowlist_0.1
- x-42c-response-allowlist_0.1

API Protection creates an **allowlist** of the valid operations and input data based on the API contract, and API Firewall enforces this configuration to all transactions, incoming requests as well as outgoing responses. Transactions containing things not described in the API definition are **automatically blocked**:

- Messages where the input or output data does not conform to the JSON schema
- Undocumented methods (POST, PUT, PATCH...)
- Undocumented error codes
- Undocumented schemas
- Undocumented query or path parameters

#2: JWT validation

JWT token validation performs a variety of checks on request tokens and blocks invalid requests

```
/api/user/info:
    get:
        x-42c-local-strategy:
            x-42c-strategy:
                protections:
                    - x-42c-jwt-validation_0.1:
                        header.name: x-access-token
                        jwk.envvar: JWK_PUBLIC_RSA_KEY
                        authorized.algorithms: [RS256, RS384]
                    # ...
```

- x-42c-jwt-validation_0.1
- x-42c-jwt-validation-ec_0.1
- x-42c-jwt-validation-hmac_0.1
- x-42c-jwt-validation-rsa_0.1

#3: Rate limitation

This protection limits how many requests API Firewall accepts from an IP address within a given time window.

```
: x-42c-request-limiter_0.1
```

```
/api/login:
    post:
        x-42c-local-strategy:
            x-42c-strategy:
                protections:
                    - x-42c-request-limiter-based-on-ip_v0.1:
                        window: 15
                        hits: 10
                        burst.enabled: true
                        burst.window: 2
                        burst.hits: 5

                    # ...
```

#4: Security headers

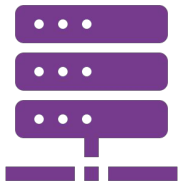
x-42c-security-headers_0.1

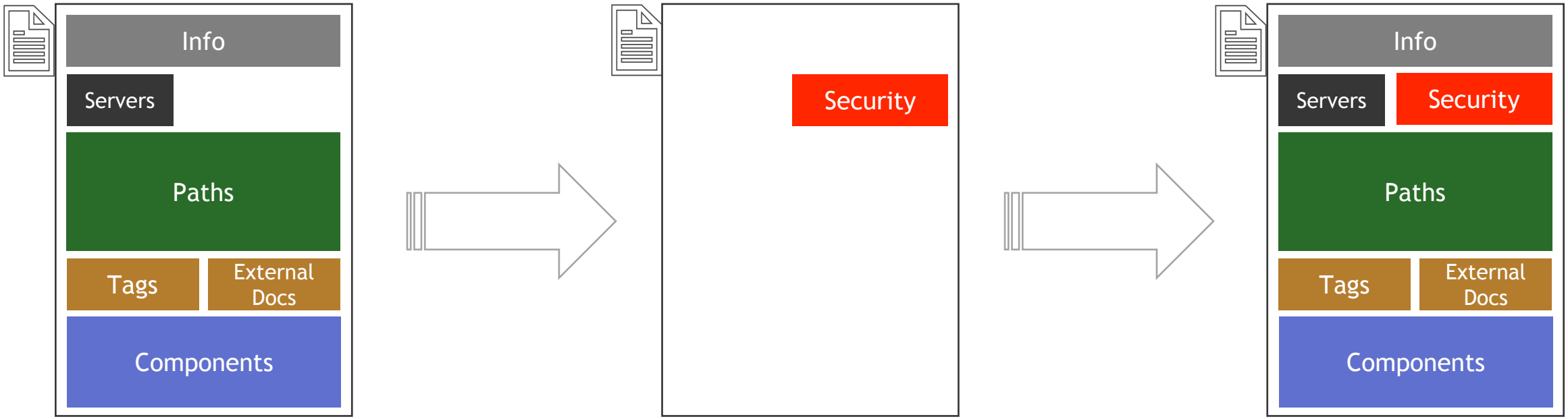
These protections on APIs control security headers either locally to specific paths, operations, responses, or alternatively to all incoming requests or outgoing responses.

```
responses:
    200:
        # ...
        x-42c-local-strategy:
            x-42c-strategy:
                protections:
                    - x-42c-security-headers_0.1:
                        hsts.max_age: 7200
                        csp.policy: default-src: 'self'; upgrade-insecure-requests;
block-all-mixed-content]
                        mode: add-replace
                    # ...
```

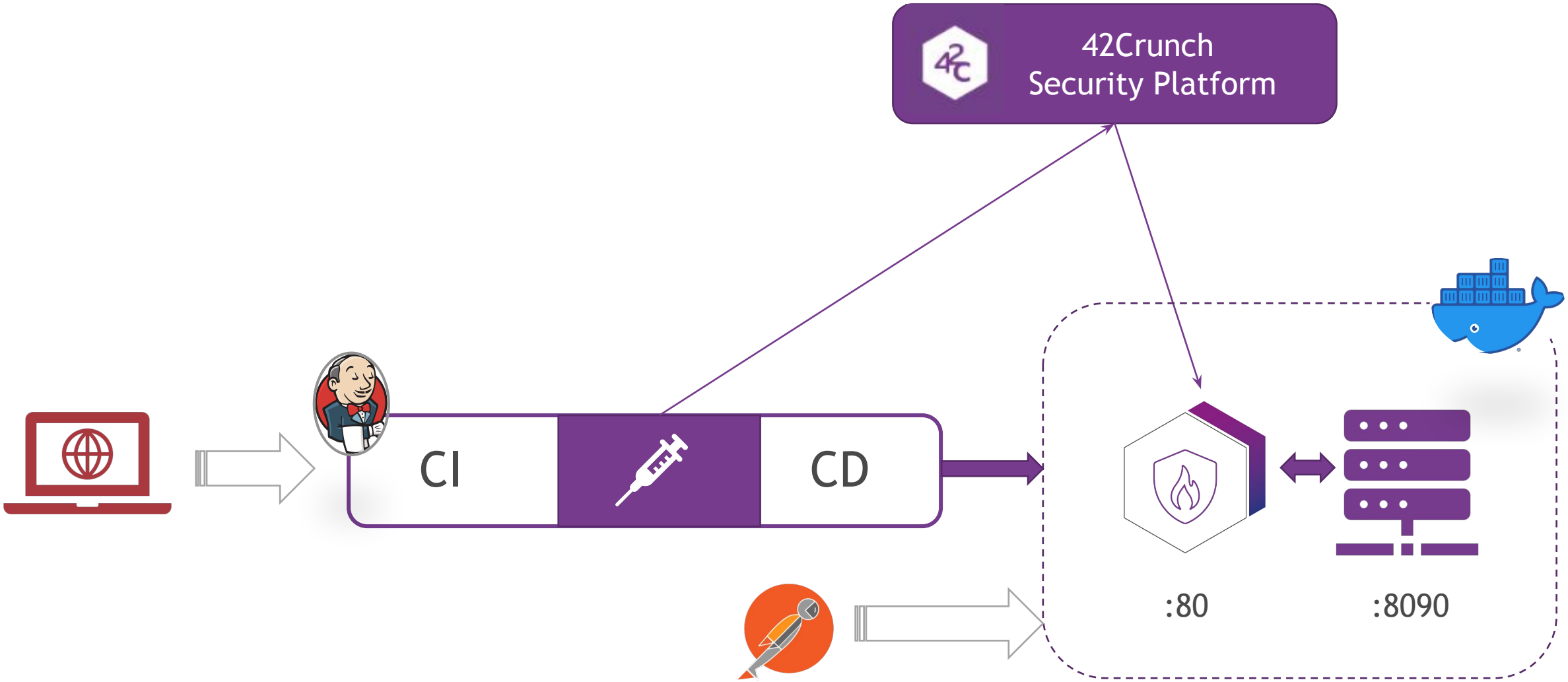
Live demo

How protection injection works with the OAS





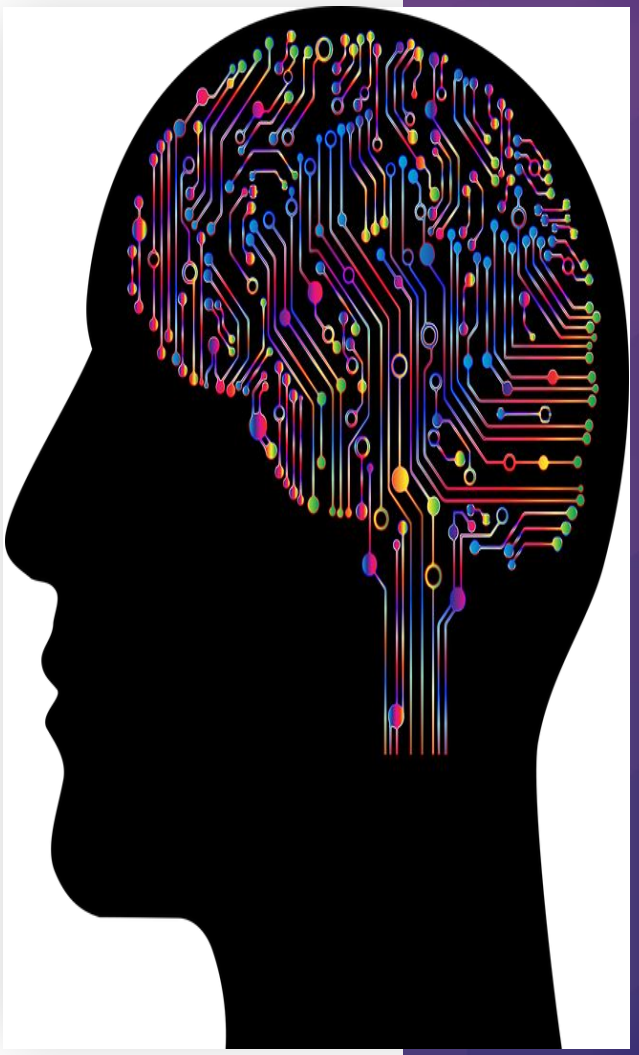
API protection injections demonstration



Question Three:

How is the OpenAPI Specification adopted/used within your organization?

1. Design first, then code
2. A mix of code and design first
3. We don’t use OpenAPI Specifications yet
4. Code first, then reverse-engineer the specification (i.e., SwaggerUI)
5. Don’t know



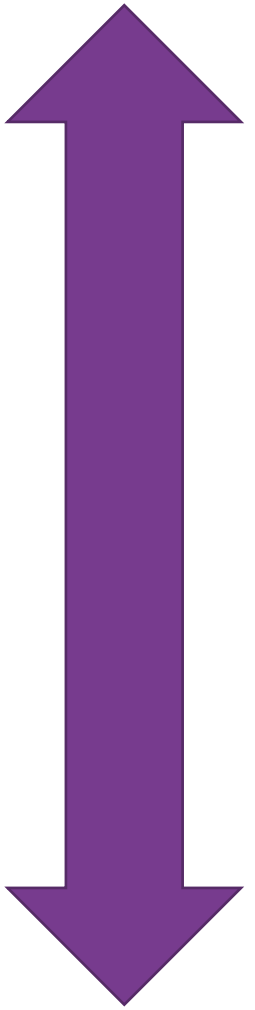
Q 3: How is the OpenAPI Specification adopted/used within your organization?

- Design first, then code — 29%
- A mix of code and design first — 45%
- We don’t use OpenAPI Specifications yet — 5.5%
- Code first, then reverse-engineer the specification (i.e. SwaggerUI) — 11.5%
- Don’t know — 9%

Use cases / Benefits

Use cases

- A Bad Thing ™ has happened !
- Emerging threats
- Legacy APIs
- Defense in depth
- Changes to policy
- Central governance
- New features
- New development teams

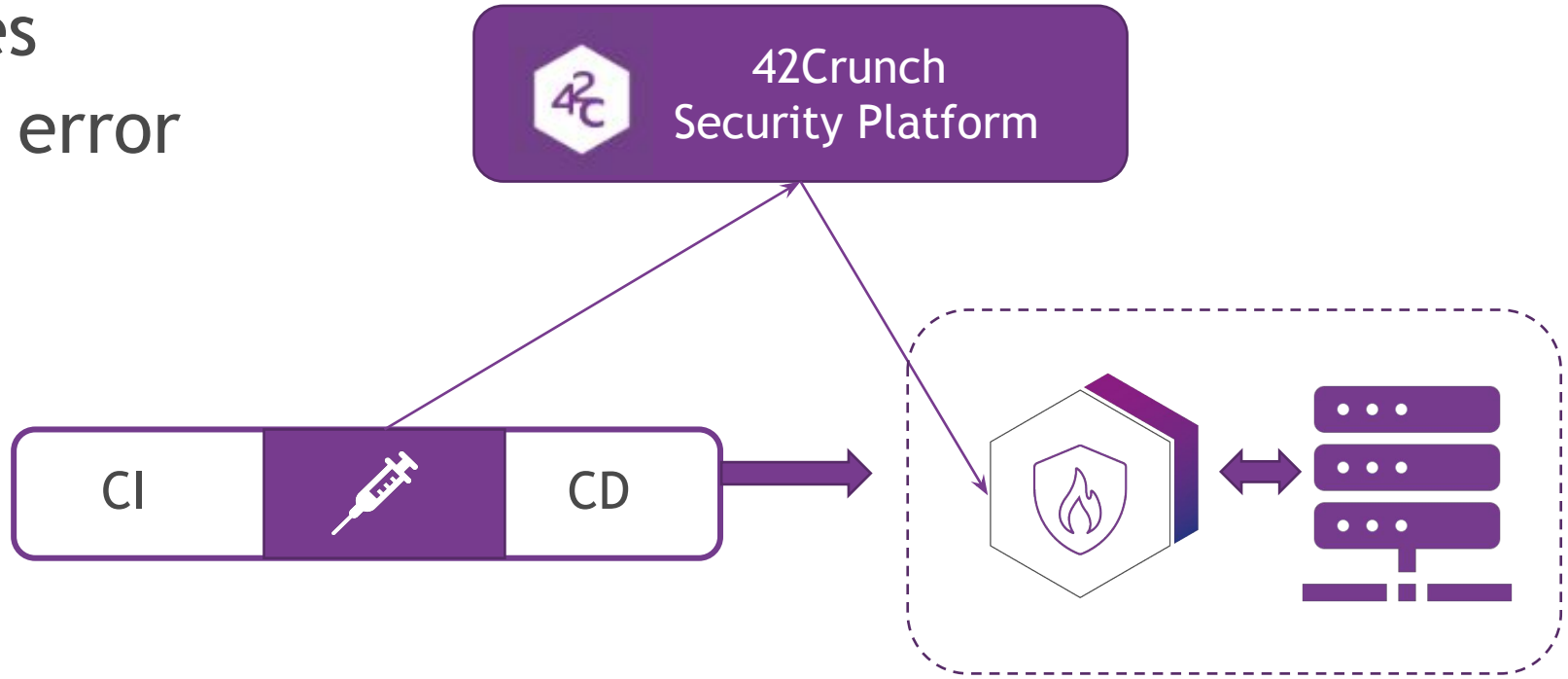


Reactive

Proactive

Benefits

- Scale
- Central governance
- Segregation of duties
- Reduction in human error



Extra Reading

Further Information

KuppingerCole API Management and Security Compass Leadership Report
https://42crunch.com/2021-kuppingercole-leadership-compass-report-for-api-management-and-security-solutions-2/

Omdia Next Gen Application Security Radar Report
https://42crunch.com/omdia-next-generation-application-security-radar-report/

APIsecurity.io Weekly Newsletter
https://apisecurity.io/

OpenAPI Editor - Free Download
https://42crunch.com/resources-free-tools/