

Apificando y securizando, que es gerundio



Seguridad en APIs del Ayuntamiento de Madrid

Quienes somos



Jesús Cuadrado
Soporte al Desarrollo
Informática Ayuntamiento de
Madrid

Daniel García (cr0hn)

Consultor y asesor
independiente en
ciberseguridad

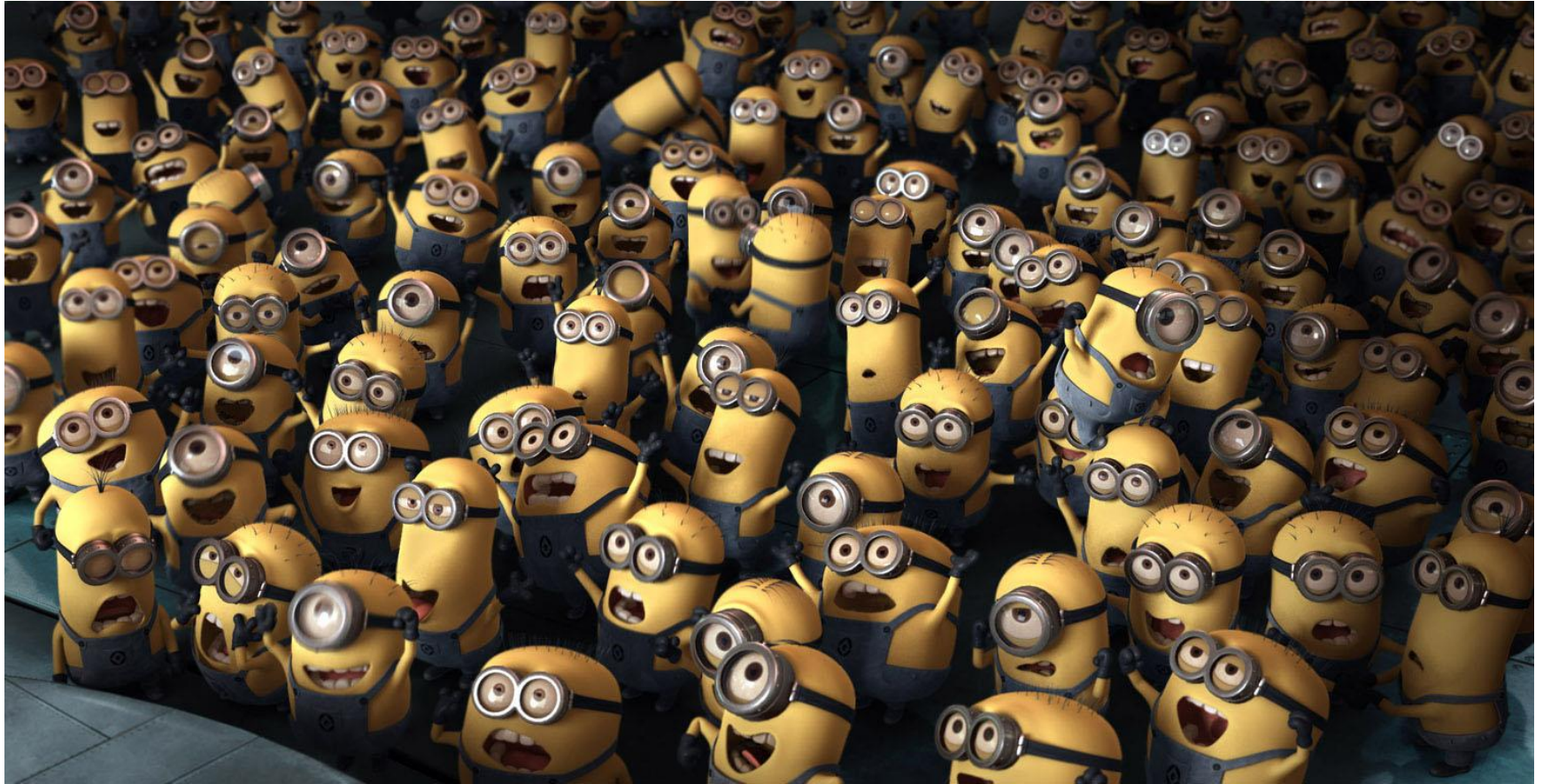


<https://www.linkedin.com/in/garciagarciadaniel/>

<https://twitter.com/ggdaniel>

Ayuntamiento de Madrid

Un entorno complejo...



... por volumetrías

Ayuntamiento de Madrid en datos



Ciudad



3,3 M
Habitantes



5 M
Población
flotante



181
Nacionalidades



540 k
Empresas



21/119/9.
422
Distritos / Barrios /
Calles



325 k
Vehículos/día en
Zonas M30



2.081/219/10.
707
Autobuses / Líneas / Paradas
(EMT)



53.427
cabezas
semáforos

Ayuntamiento



524/156
Trámites/Servicios al
ciudadano/empresa
en sede



99
Oficinas de
Atención y
Asistencia



15/5/8
Áreas de Gobierno y
Delegadas/OOAA/Empresas
municipales y mixtas



901
Sedes
municipales



30.000
Empleados



> 5.000
Empleados diarios
teletrabajo



> 6.000
Líneas móviles
corporativas

Tecnología



410
Servidores de
aplicaciones



60 TB
Tráfico mensual
datos



1.546
Servidores
virtuales



83
Servidores
físicos



5M
Paquetes / hora FW
perimetral



20 M
Eventos / hora
SIEM



865
Bases de
Datos

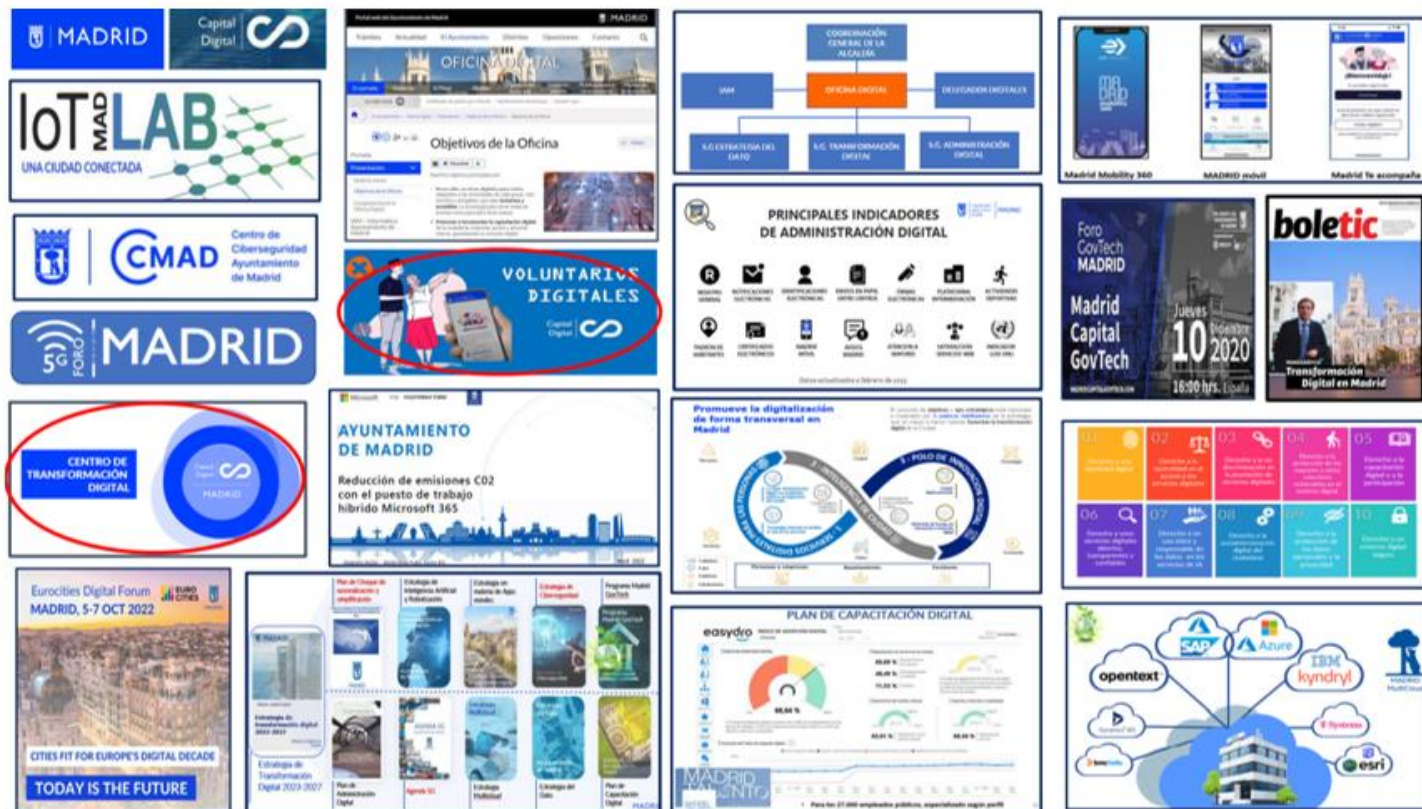


4437
TB de
almacenamiento

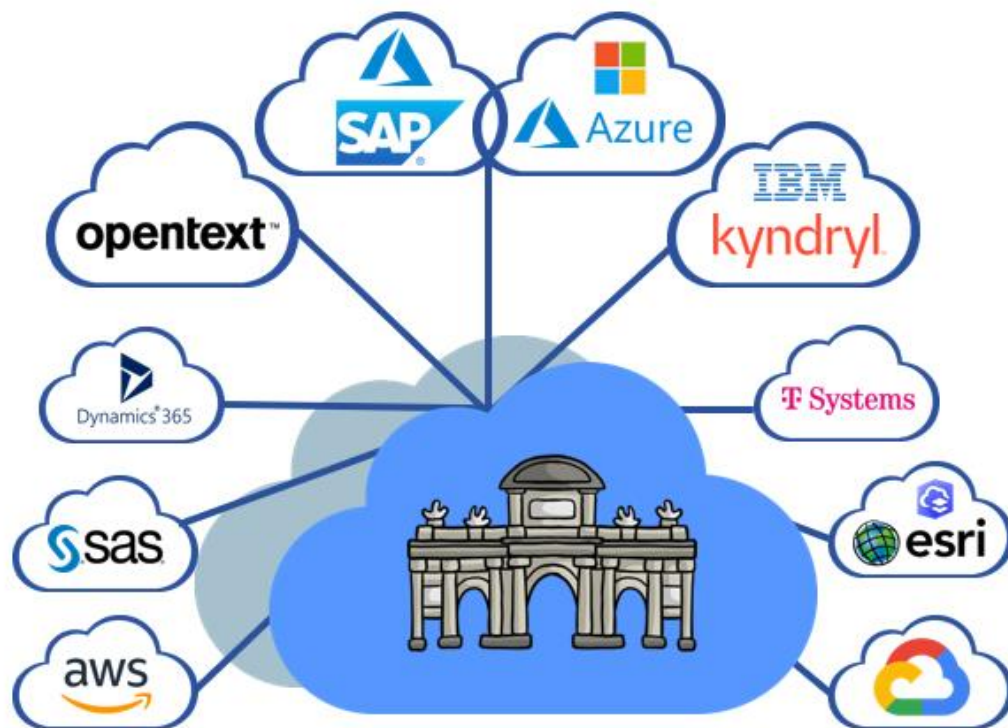


34 k
Buzones correo
corporativos

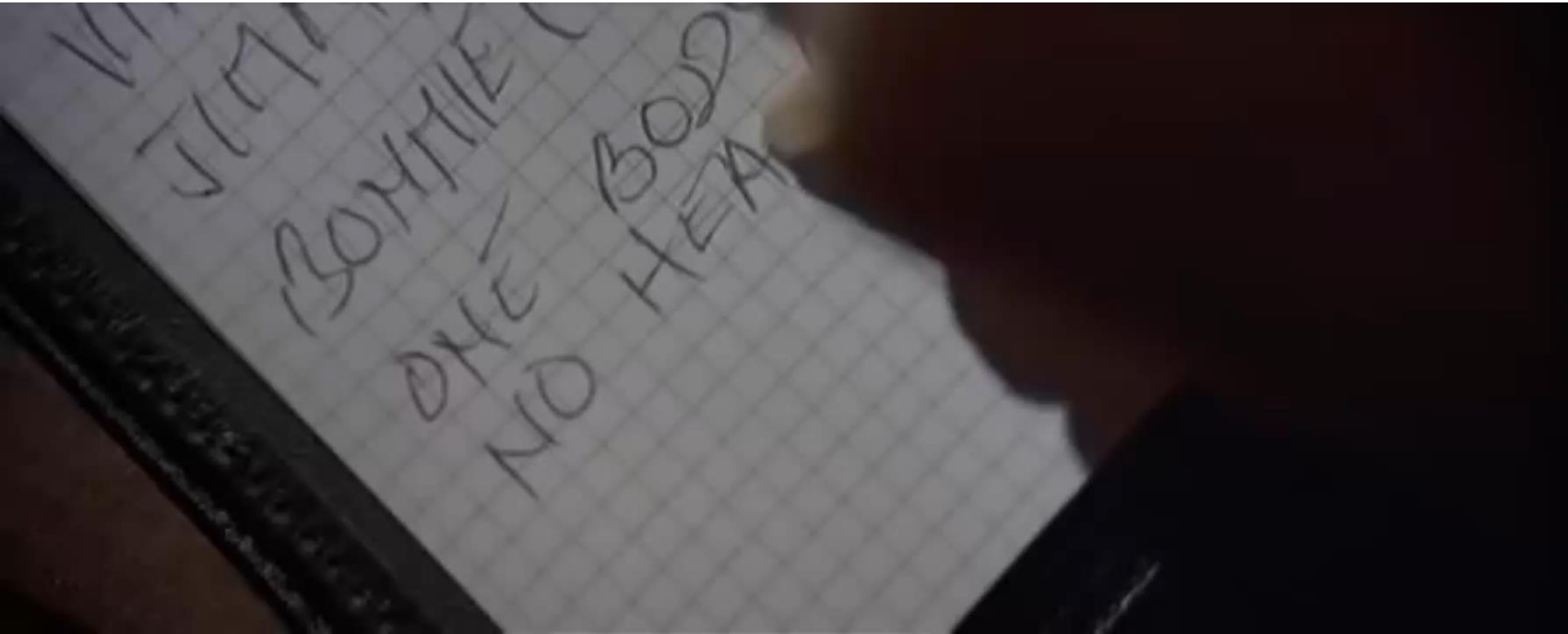
...por iniciativas



...por infraestructura



...mi trabajo (no soy hacker)



Situación de partida



01

Entornos INTERNET / INTRANET

02

Http y red plana

03

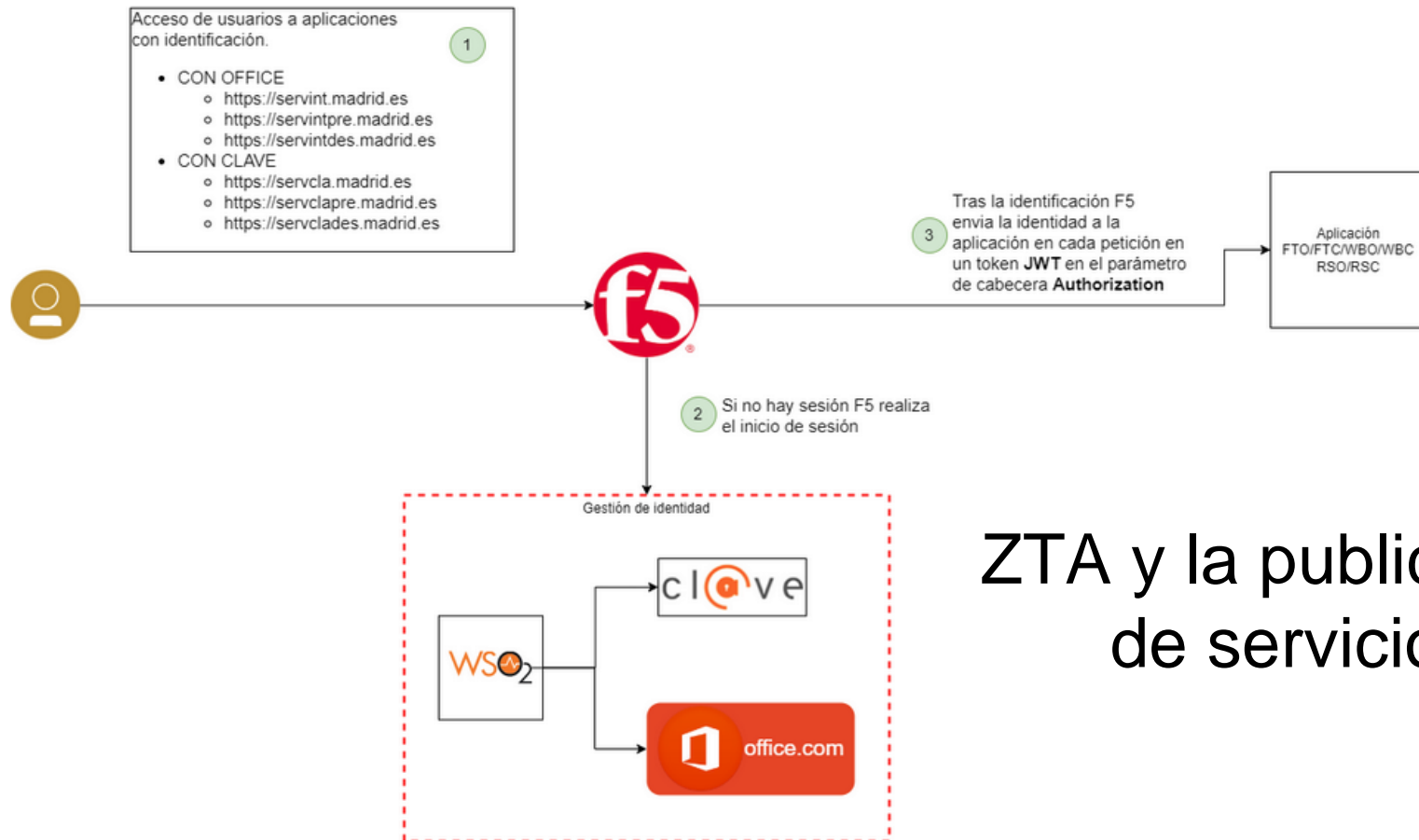
Autenticación y autorización de usuarios casera

04

Algoritmos débiles

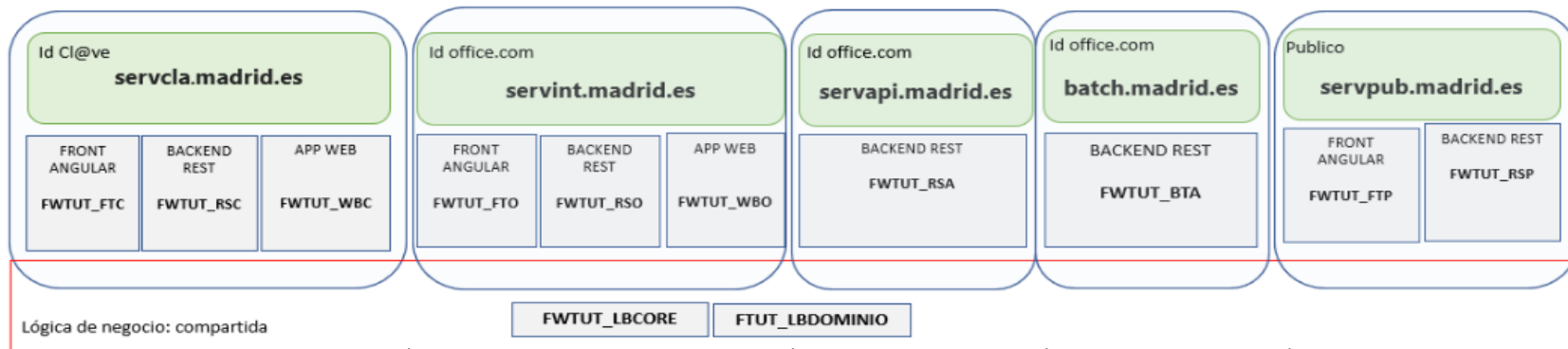
1.1. Identificación de usuarios y comunicación FT → RS y WB con f5-APM

El ciclo principal de identificación de usuarios está controlado por f5-APM de forma que las aplicaciones no tienen que hacer nada adicional.



ZTA y la publicación de servicios

Arquitectura framework IAM



Componentes de ciudadano que se autentican usando cl@ve:

*FTC - Frontales Angular
RSC – Servicios Rest que sirven a peticiones de los frontales
WBC – Aplicación Web que necesita de autenticación de cl@ve*

Componentes de ciudadano que se autentican usando office.com

*FTO - Frontales Angular
RSO – Servicios Rest que sirven a peticiones de los frontales
WBO – Aplicación Web que necesita de autenticación de office.com*

API que los componentes se exponen entre sí para realizar integraciones

Componentes Batch

Componentes Públicos (usuario/contraseña) o sin autenticación

*Componentes de Librería reutilizables LOB**

DevSecOps como plataforma

Presentación del framework de desarrollo

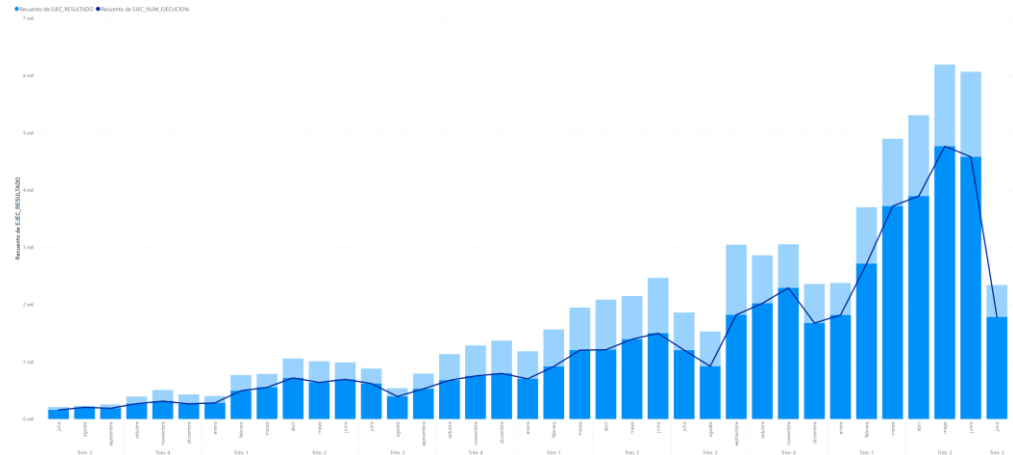


Hace 3 años

- Despliegues manuales
- 72h SLA
- Cultura SISOps

2023

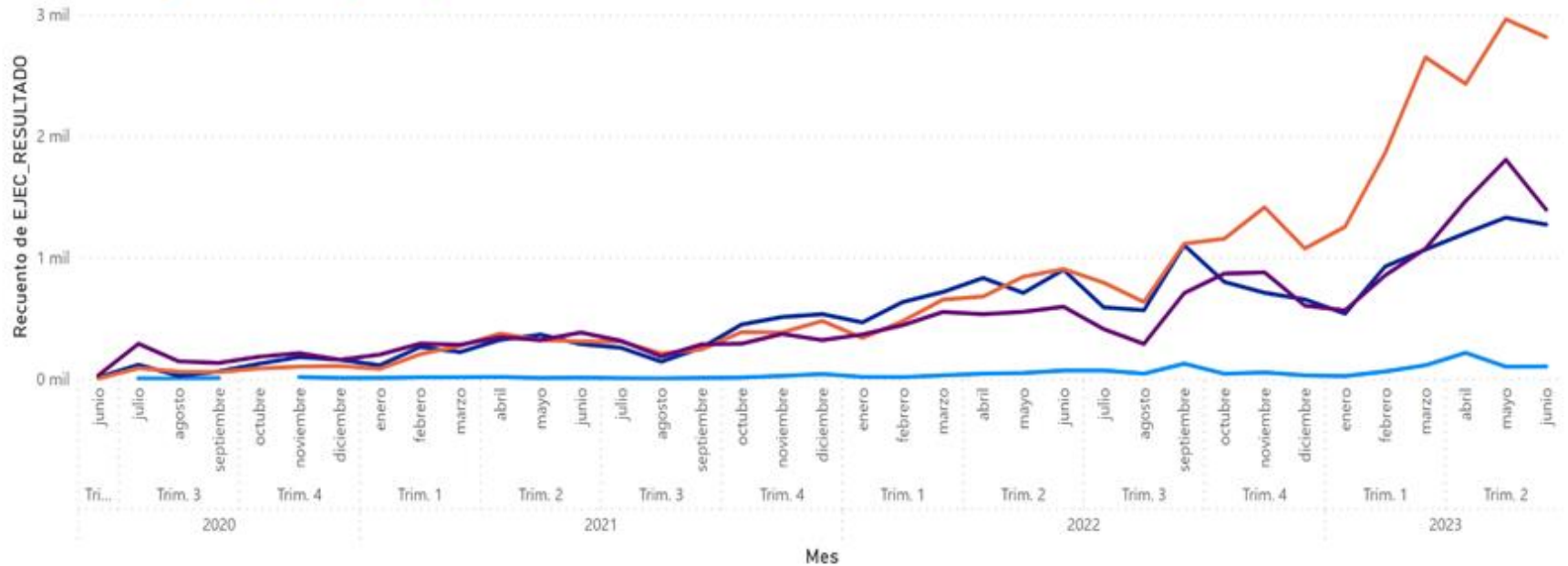
- Despliegues x60
- No hay SLA
- Cultura DevSecOps + GitOps



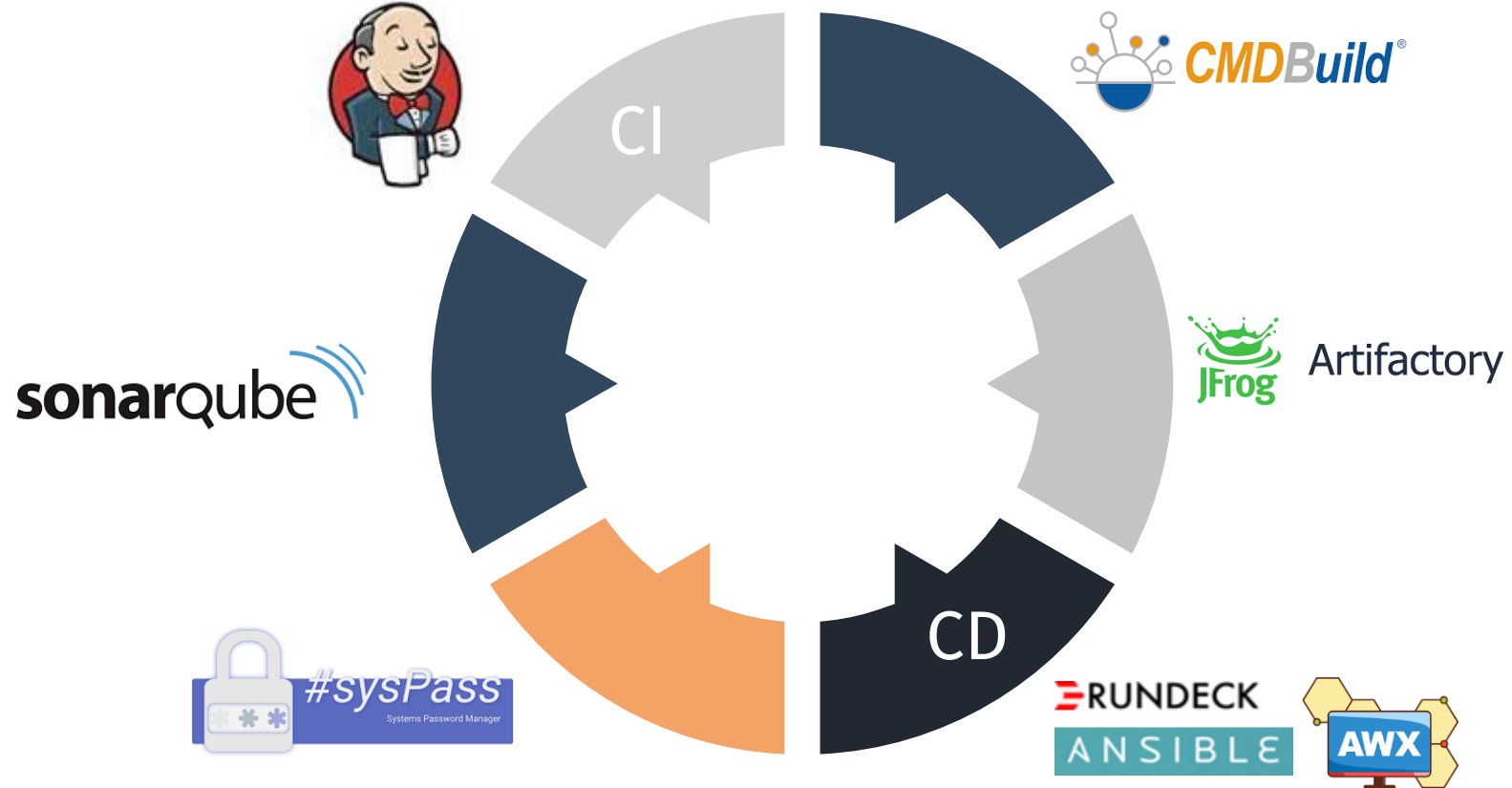
DevSecOps Evolución

Recuento de EJEK_RESULTADO por Año, Trimestre, Mes y EJEK_RESULTADO

EJEK_RESULTADO ● ABORTED ● FAILURE ● SUCCESS ● UNSTABLE



DevSecOps con medios propios




Automatización

 RUNDECK

 PROJECTS

 DASHBOARD

 JOBS

 NODES

 COMMANDS

 ACTIVITY

DEPARTAMENTO DE TECNOLOGIA

Jobs (119)

Filter > Expand All Collapse All

Job Actions >

> ALMACENAMIENTO

> Actualizacion Plataforma RUNDECK

> Arquitectura Ligera Wildfly

> BATCH

> BBDD

> ELK

> Infraestructura

> KAFKA

> LEGACY

> Mantenimiento

> Migracion WAS 61

> Plataforma WAS



**APIs la asignatura
pendiente**

Problemas con las APIs

Auditar

No se sabe
auditar



Protección

No se protege igual



Conocimiento

Nadie sabe lo que tiene



Contratos

No existen contratos
ni se saben generar



Nuestras APIs

Qué sabemos de nuestras APIs



Nuestras APIs – Cuántas temenos?



Nuestras APIs – Qué sabemos de ellas?



Nuestras APIs – contratos (1/2)

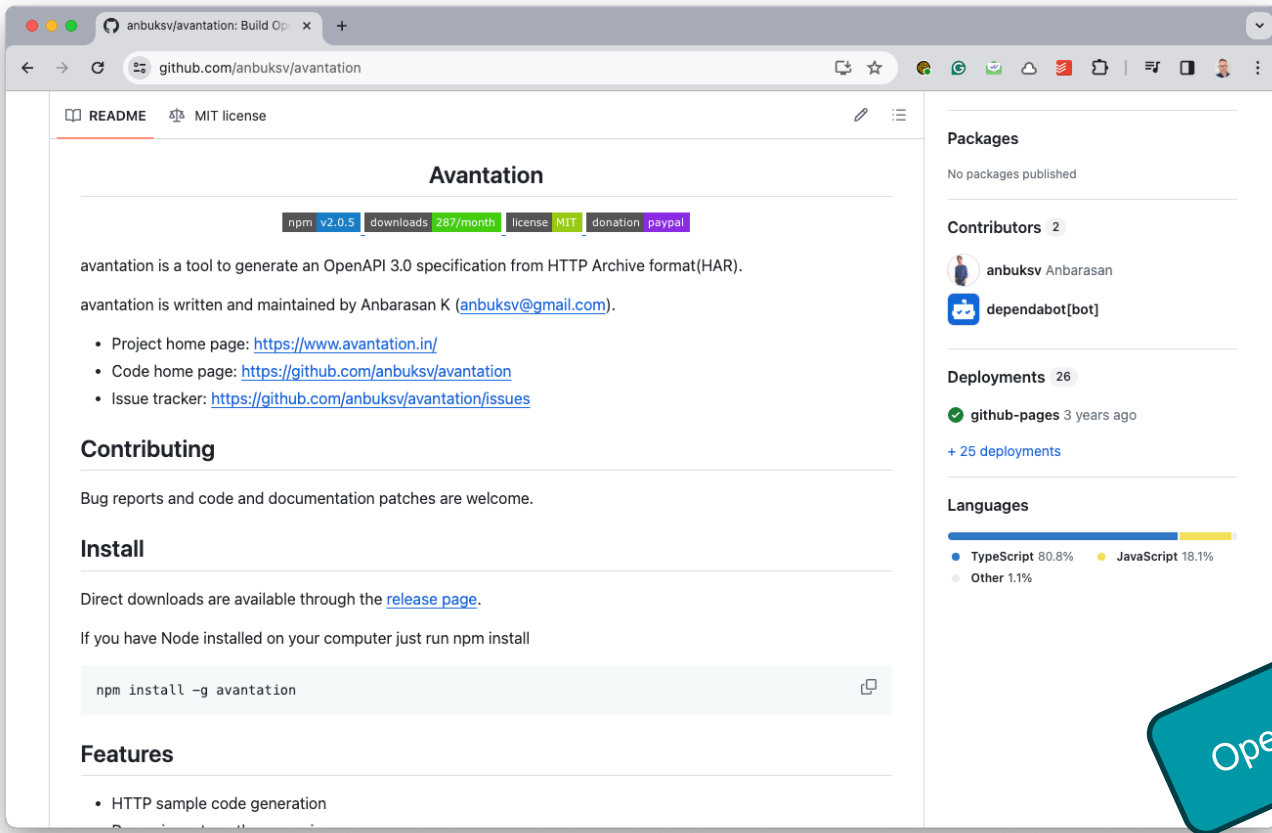


Nuestras APIs – contratos (2/2)

- RAML
- Swagger
- OpenAPI
- PostMan



Nuestras APIs – generando contratos (1/4)



The screenshot shows the GitHub repository page for 'anbuksv/avantation'. The page includes a README section with the following content:

Avantation

npm v2.0.5 | downloads 287/month | license MIT | donation paypal

avantation is a tool to generate an OpenAPI 3.0 specification from HTTP Archive format(HAR).

avantation is written and maintained by Anbarasan K (anbuksv@gmail.com).

- Project home page: <https://www.avantation.in/>
- Code home page: <https://github.com/anbuksv/avantation>
- Issue tracker: <https://github.com/anbuksv/avantation/issues>

Contributing

Bug reports and code and documentation patches are welcome.

Install

Direct downloads are available through the [release page](#).

If you have Node installed on your computer just run npm install

```
npm install -g avantation
```

Features

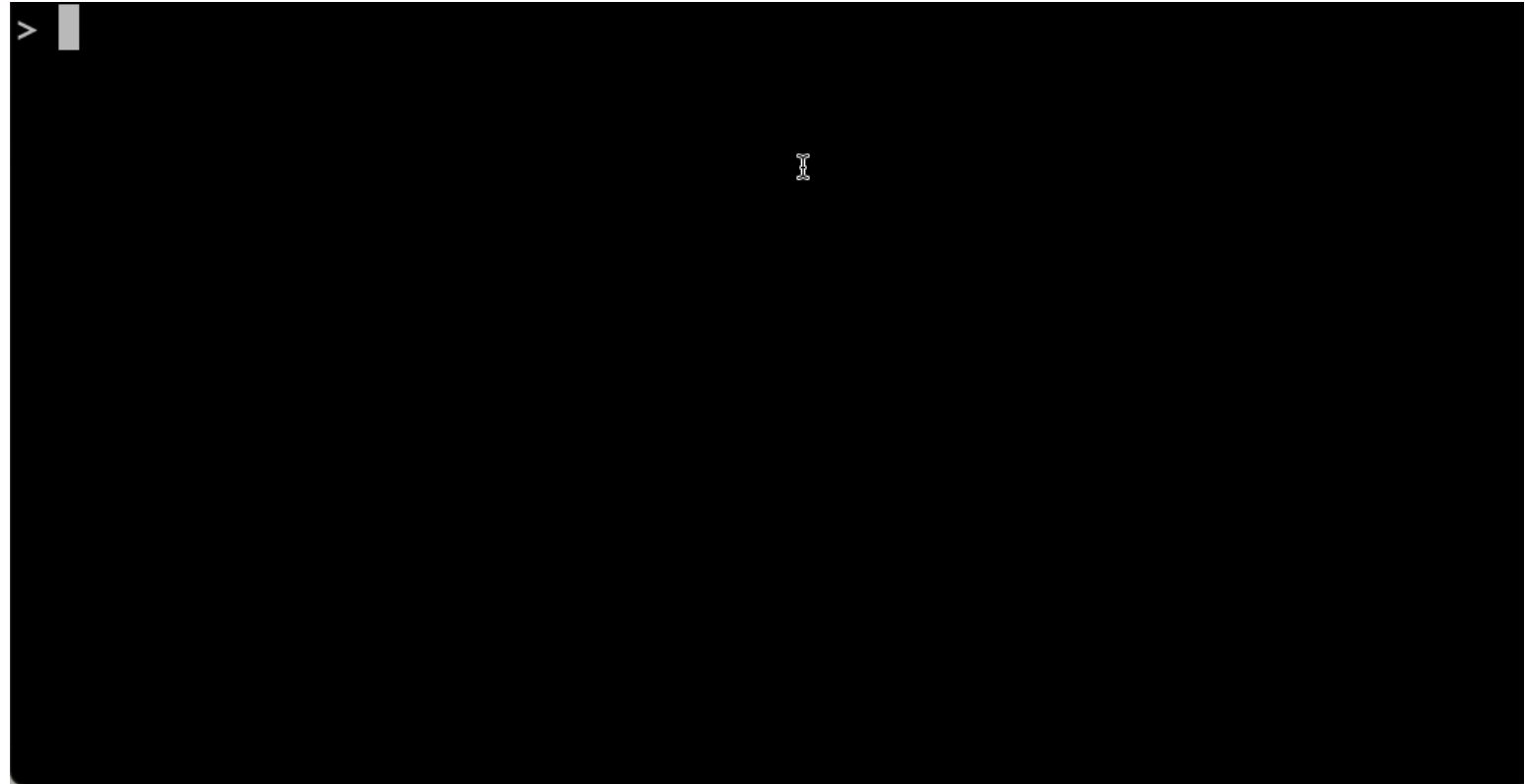
- HTTP sample code generation

The right sidebar shows repository statistics:

- Packages:** No packages published
- Contributors:** 2 (anbuksv Anbarasan, dependabot[bot])
- Deployments:** 26 (github-pages 3 years ago, +25 deployments)
- Languages:** TypeScript 80.8%, JavaScript 18.1%, Other 1.1%

Open Source

Nuestras APIs – generando contratos (1/4)



Nuestras APIs – generando contratos (2/4)

The screenshot shows the GitHub repository for HttpShark. The README file is open, displaying the project's overview and architecture. The architecture diagram illustrates the flow from a central 'Capture' point to two different output methods: 'Http Dump' and 'Command Line'.

HttpShark

Overview

HttpShark is an application that captures packets from a network interface, collects and correlates HTTP requests and HTTP responses, and produces HAR files containing the HTTP transactions.

Architecture

The application architecture is as follows:

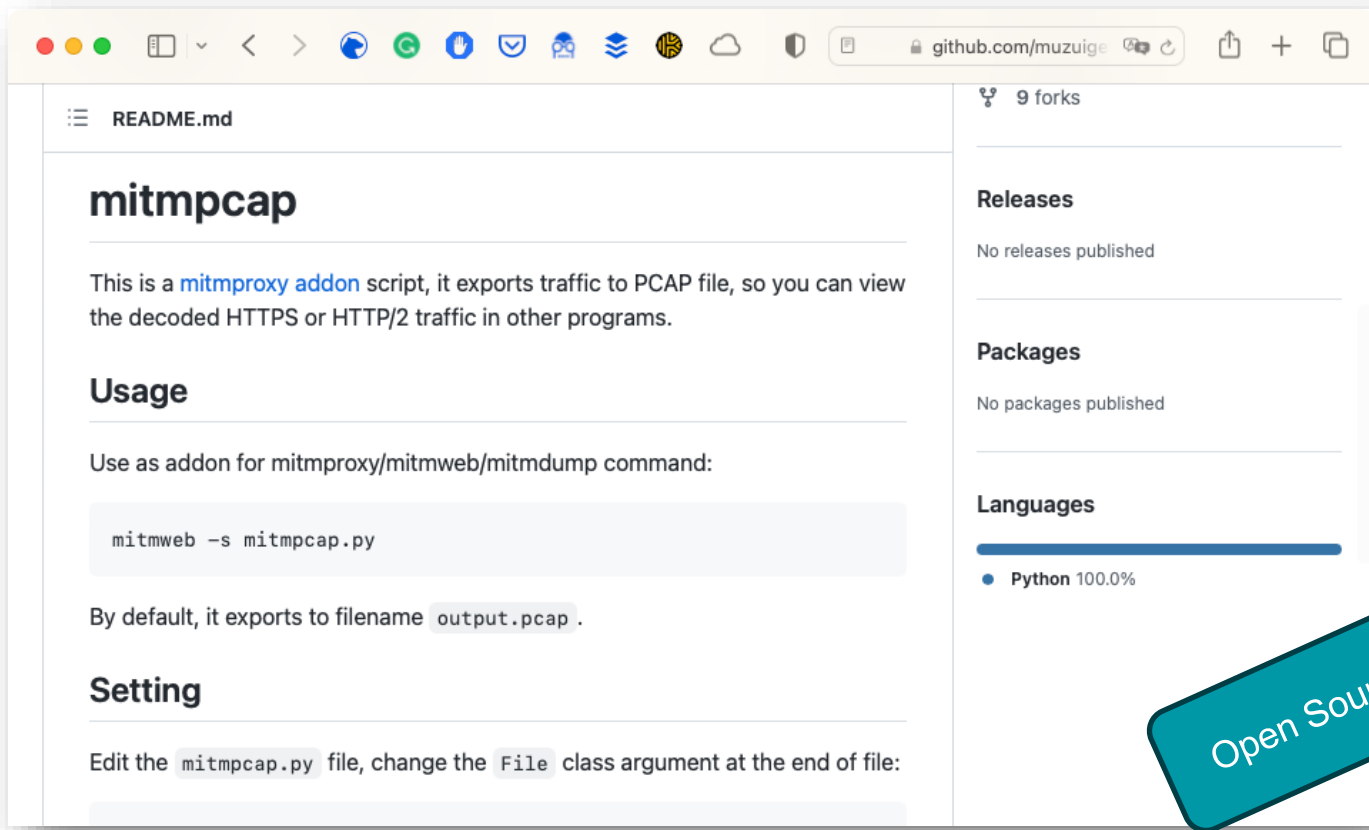
```
graph TD; Capture{Capture} -- httpdump --> HttpDump[Http Dump]; Capture -- tshark --> CommandLine[Command Line];
```

On the right side of the repository page, the language distribution is shown:

- Go 96.4%
- Shell 3.2%
- Python 0.4%

Open Source

Nuestras APIs – generando contratos (3/4)



The screenshot shows a web browser window displaying the GitHub repository page for `mitmpcap`. The browser's address bar shows the URL `github.com/muzuige`. The repository page has a light gray header with the file name `README.md` on the left and `9 forks` on the right. The main content area is divided into two columns. The left column contains the README text, and the right column contains repository statistics.

mitmpcap

This is a [mitmproxy](#) [addon](#) script, it exports traffic to PCAP file, so you can view the decoded HTTPS or HTTP/2 traffic in other programs.

Usage

Use as addon for mitmproxy/mitmweb/mitmdump command:

```
mitmweb -s mitmpcap.py
```

By default, it exports to filename `output.pcap`.

Setting

Edit the `mitmpcap.py` file, change the `File` class argument at the end of file:

Releases

No releases published

Packages

No packages published

Languages

Python 100.0%

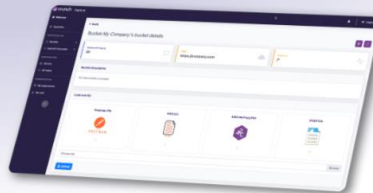
Open Source

Nuestras APIs – generando contratos (4/4)

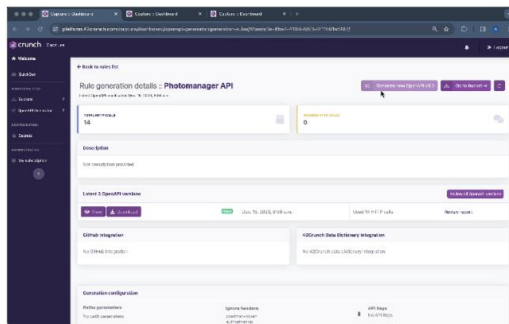
API Capture

Automation of OpenAPI Contracts & Security Test Configurations

API Capture Datasheet



API Capture automates the generation of OpenAPI contracts and API security testing configurations from Postman collections and API traffic. This innovative approach expedites testing timelines, minimizes manual effort, and ensures highly secure and scalable APIs. This results in improved compliance controls and the reduction of development times and delivery costs for secure APIs.



<https://42crunch.com/api-capture/>

Nuest

platform.dev.42crunch.com/capture/dashboard/quickgen/files/

Slack JIRA BambooHR New Tab Gmail AWS Confluence API Builder Platfor... Google Cloud con... Jira Capture

42crunch Capture

Welcome

QuickGen

FUNCTIONALITIES

Buckets >

OpenAPI Generator >

CONFIGURATION

Secrets

ADMINISTRATION

My subscription

Add files

Reset Next step →

In order to continue to the next step you must have set a base path and at least one source file uploaded.

Base path pattern


/*

You can use * wildcard. Example: /api/account/* or /api/*. The fixed part will be used as the OpenAPI base path.

Save base path


Load new file

Postman File



POSTMAN

HAR File



har file

Choose file Browse

Upload

Total HTTP calls: 0

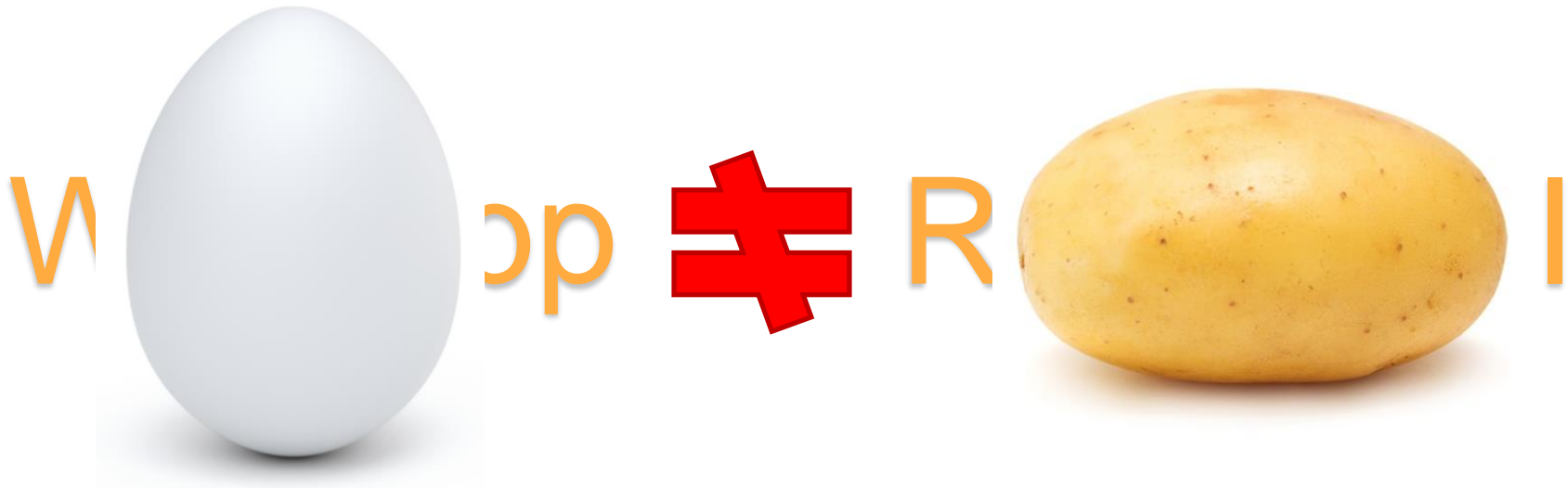
4)

Cómo se protegen

Qué sabemos de nuestras APIs

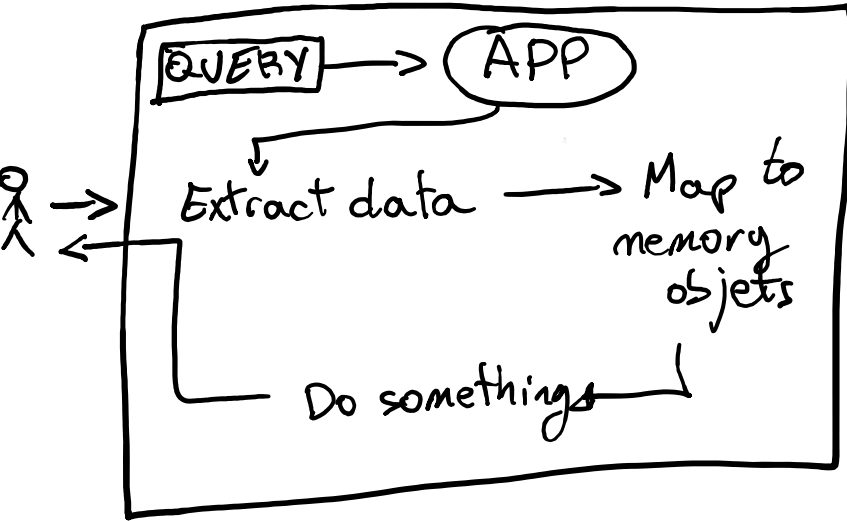


Cómo se protegen – Web App vs REST API (1/3)

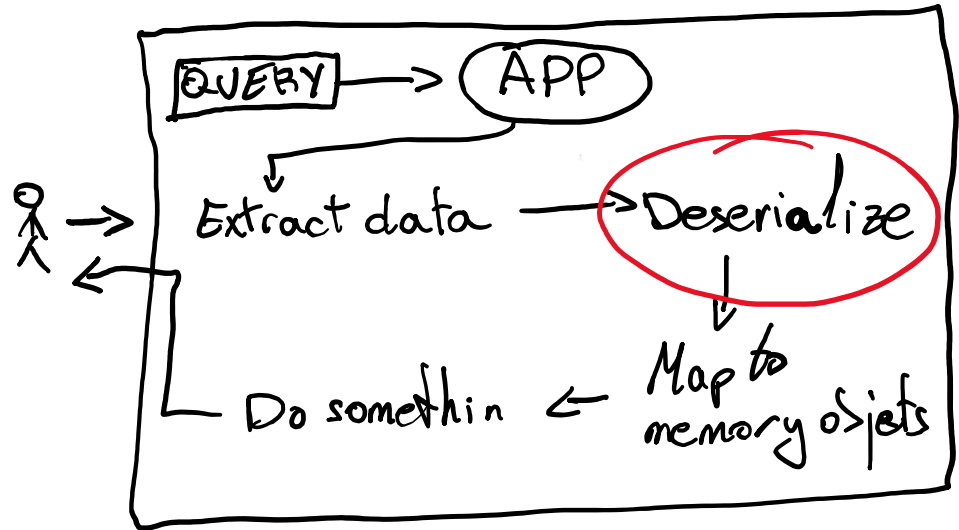


Cómo se protegen – Web App vs REST API (2/3)

Web App



rest API



Cómo se protegen – Web App vs REST API (3/3)

Web App

- Usa HTTP
- Usa HTML
- Se renderiza en backend
- La respuesta está pensada para el navegador

rest API

- Usa HTTP
- Usa un mecanismo de serIALIZACIÓN
- No necesita ser renderizado
- La respuesta está pensada para construir otro software

Cómo se protegen – WAF Tradicional (1/3)

No fueron diseñados para entender los detalles de implementación y nuevas formas de trabajo de las APIs REST

Cómo se protegen – WAF Tradicional (2/3)

```
SecRule REQUEST_COOKIES:!REQUEST_COOKIES:/ utm/|REQUEST_COOKIES NAMES|ARGS NAMES|ARGS|XML:/*  
"@rx (?i)alter[\s\v]*?[0-9A-Z_a-z]+.*?char(?:acter)?[\s\v]+set[\s\v]+[0-9A-Z_a-z]+|[\s\v]`  
l(?:.*?[\s\v]*?waitfor[\s\v]+(?:time|delay)[\s\v]+[\s\v"]|[:.*?[\s\v]*?goto)" \  
  "id:942240,\ \  
  phase:2,\ \  
  block,\ \  
  capture,\ \  
  t:none,t:urlDecodeUni,\ \  
  msg:'Detects MySQL charset switch and MSSQL DoS attempts',\  
  logdata:'Matched Data: %{TX.0} found within %{MATCHED_VAR_NAME}: %{MATCHED_VAR}',\  
  tag:'application-multi',\  
  tag:'language-multi',\  
  tag:'platform-multi',\  

```

Cómo se protegen – WAF Tradicional (3/3)

```
{
  "userName": "Jhon",
  "surName": "Deer",
  "personaInfo":
  [
    {
      "Addresses": [
        {
          "name": "work",
          "address": "Puerta del sol 1, Madrid"
        },
        {
          "name": "home",
          "address": "Calle Oña 10, Madrid"
        }
      ],
      "mainAddress": "home",
      "protectPersonalInfo": true
    }
  ]
}
```

Objeto JSON de-serializado

```
"{"\\\"userName\\\":\\\"Jhon\\\",
\\\"surName\\\":\\\"Deer\\\",\\\"p
ersonaInfo\\\":[{\\\"Address
es\\\":[{\\\"name\\\":\\\"work\\\"
,\\\"address\\\":\\\"Puerta
del sol 1, Madrid\\\"},{\\\"
name\\\":\\\"home\\\",\\\"adres
s\\\":\\\"Calle Oña 10, Madrid\\\"}],\\\"mainAdd
ress\\\":\\\"home\\\",\\\"protec
tPersonalInfo\\\":true}]]\"
```

Objeto JSON serializado

Cómo se protegen – WAF orientados a API REST (1/3)

- Tienen una fase de descubrimiento
- Comprende los comportamientos de las API
- Entiende los datos serializados y sus conceptos de herencia
- Sabe cómo tratar cada tipo de datos
- Fueron diseñados para las APIs REST

Cómo se protegen – WAF orientados a API REST (2/3)

Machine Learning

- + Plug and Play
- + Fácil arranque
- + Fácil de usar
- Falsos positivos
- Periodo de aprendizaje
- Se necesita MITM

API definition

- + No necesita MITM
- + No tiene falsos positivos
- + Reglas precisas
- Definición de API necesaria
- Arranque más complejo

Cómo se protegen – WAF orientados a API REST (3/3)


wallarm



noname



SALT

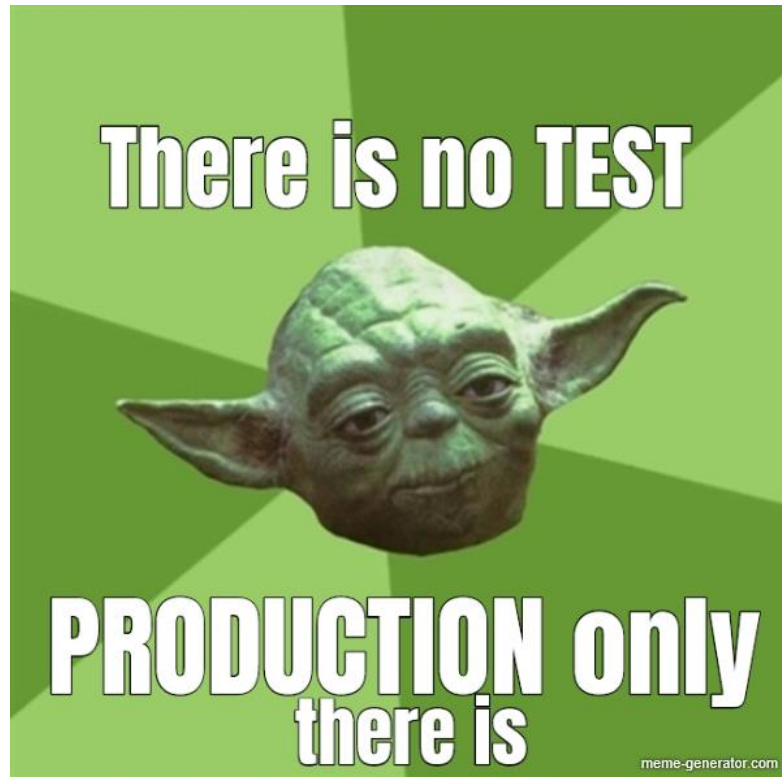
imperva



crunch

Cómo se testean

Cómo probamos nuestras APIs



Cómo se testean – Problema 1

Necesitamos tener una definición la APIs que
queremos probar

Cómo se testean – Problema 2

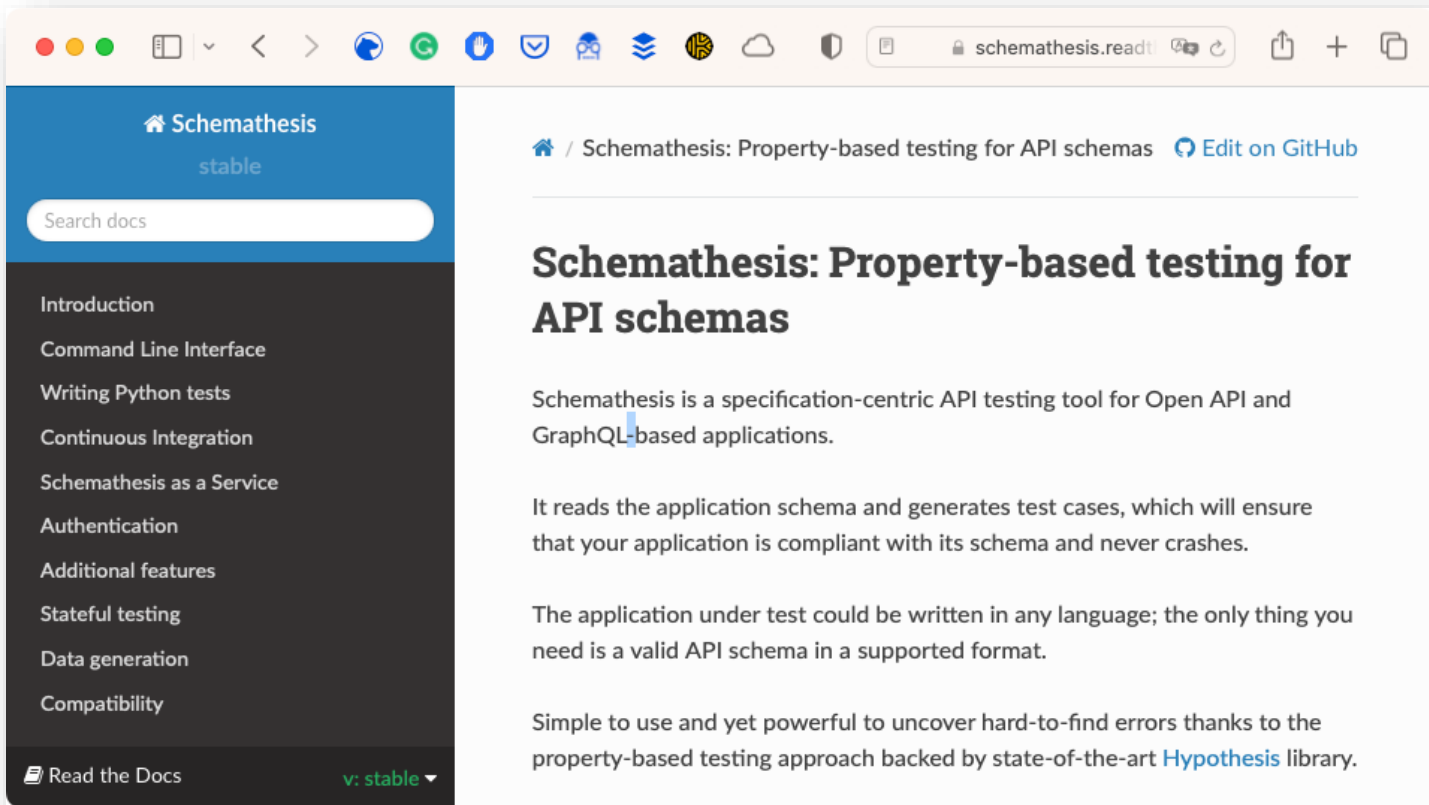
Las pruebas tienen que poder automatizarse
y ejecutarse de forma desatendida

Cómo se testean – Problema 3

Tiene que ejecutarse en un tiempo razonable

Cómo se testean – Schemathesis (1/2)

Realiza pruebas basadas en *property testing*



The screenshot shows a web browser window displaying the Schemathesis documentation on Read the Docs. The browser's address bar shows 'schemathesis.readthedocs.io'. The page has a blue header with the Schemathesis logo and the word 'stable'. A search bar is located below the header. On the left, a dark sidebar contains a list of navigation links: Introduction, Command Line Interface, Writing Python tests, Continuous Integration, Schemathesis as a Service, Authentication, Additional features, Stateful testing, Data generation, and Compatibility. At the bottom of the sidebar, there is a 'Read the Docs' button and a version selector set to 'v: stable'. The main content area has a breadcrumb trail: Home / Schemathesis: Property-based testing for API schemas, followed by a link to 'Edit on GitHub'. The main heading is 'Schemathesis: Property-based testing for API schemas'. The introductory text states: 'Schemathesis is a specification-centric API testing tool for Open API and GraphQL-based applications. It reads the application schema and generates test cases, which will ensure that your application is compliant with its schema and never crashes. The application under test could be written in any language; the only thing you need is a valid API schema in a supported format. Simple to use and yet powerful to uncover hard-to-find errors thanks to the property-based testing approach backed by state-of-the-art Hypothesis library.'

🏠 Schemathesis
stable

Search docs

Introduction
Command Line Interface
Writing Python tests
Continuous Integration
Schemathesis as a Service
Authentication
Additional features
Stateful testing
Data generation
Compatibility

📖 Read the Docs v: stable ▼

🏠 / Schemathesis: Property-based testing for API schemas [Edit on GitHub](#)

Schemathesis: Property-based testing for API schemas

Schemathesis is a specification-centric API testing tool for Open API and GraphQL-based applications.

It reads the application schema and generates test cases, which will ensure that your application is compliant with its schema and never crashes.

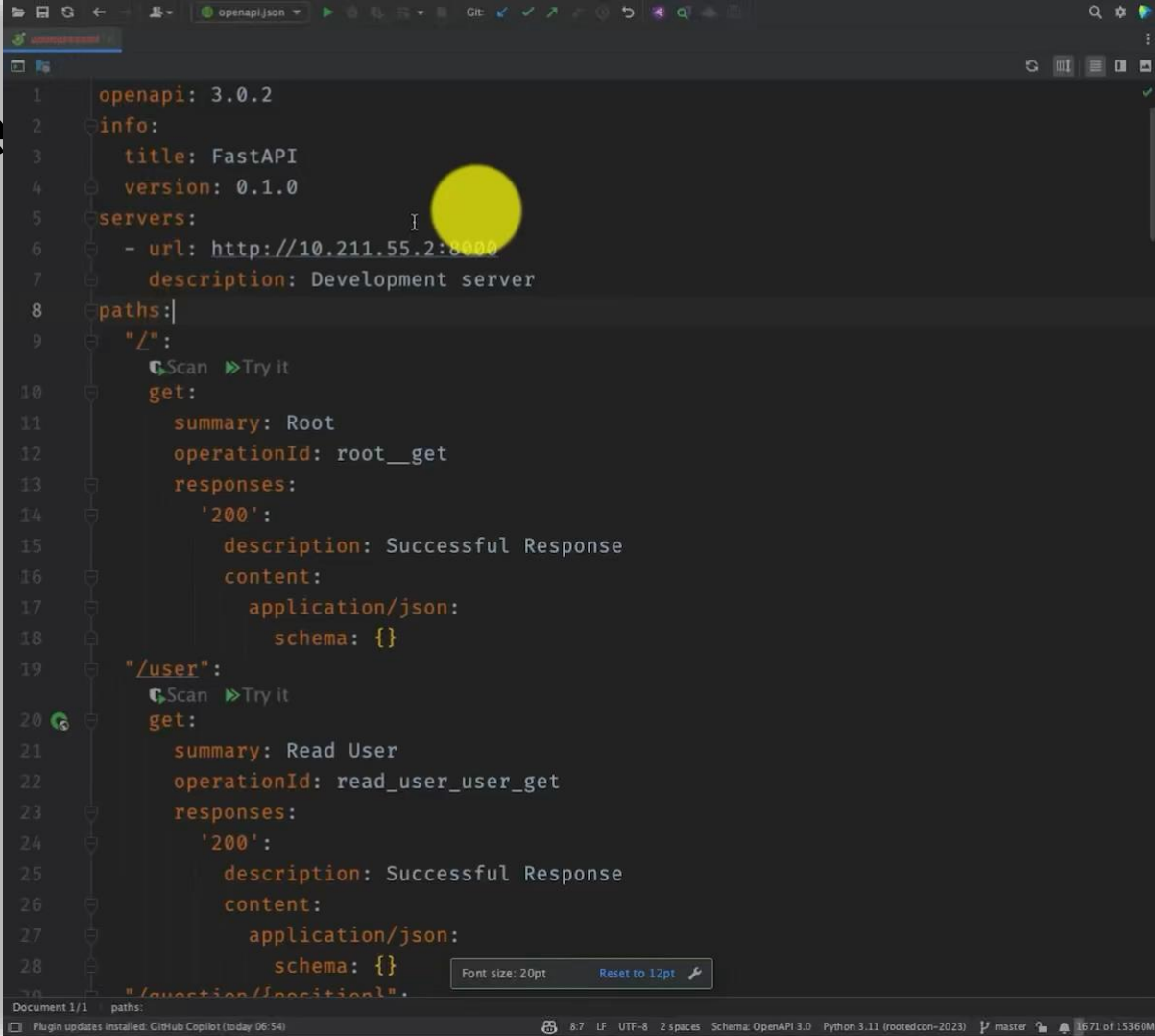
The application under test could be written in any language; the only thing you need is a valid API schema in a supported format.

Simple to use and yet powerful to uncover hard-to-find errors thanks to the property-based testing approach backed by state-of-the-art [Hypothesis](#) library.

```
> uvicorn app.main:app --reload --host 0.0.0.0
```

4

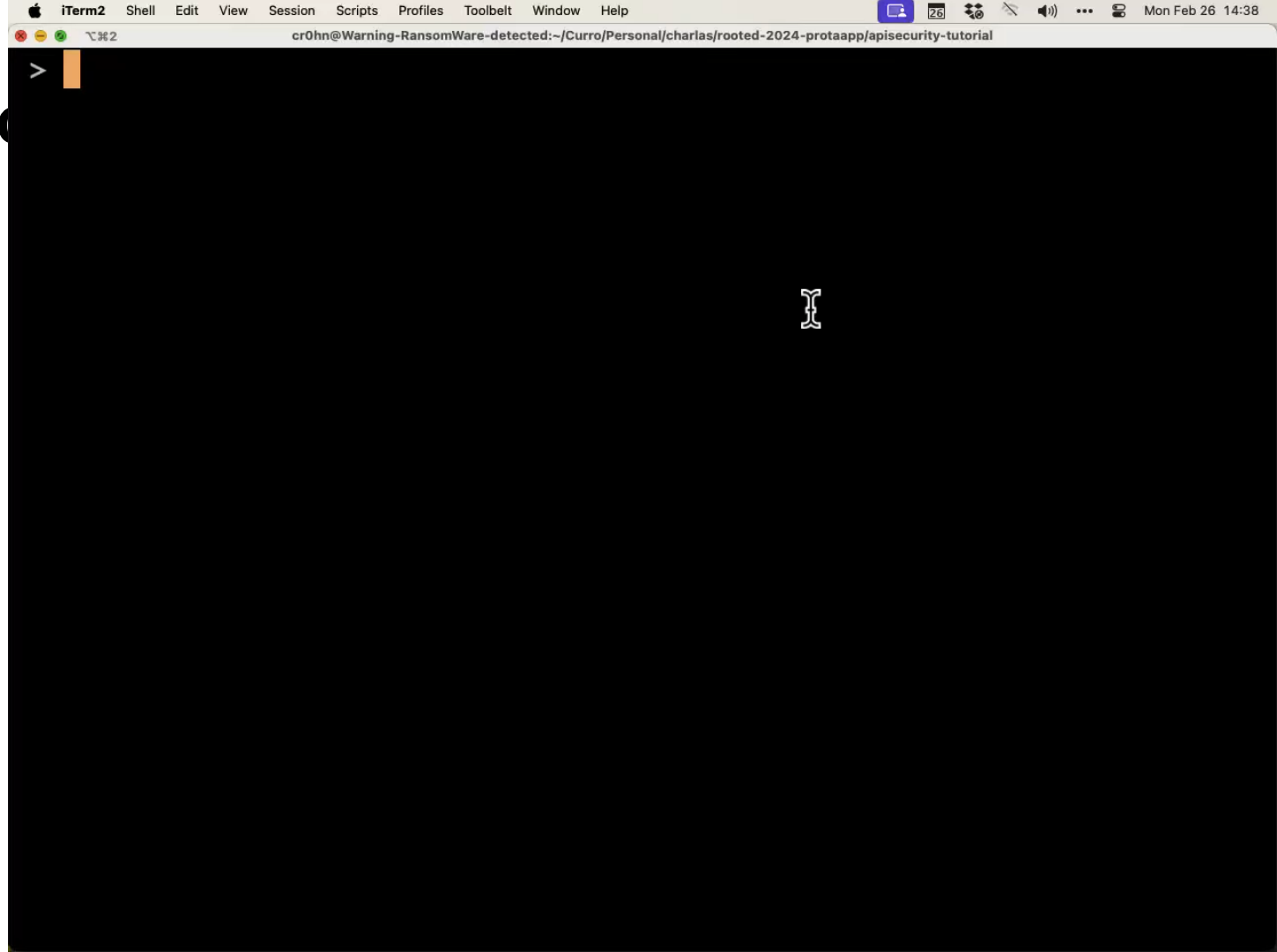
Cómo se



```
1  openapi: 3.0.2
2  info:
3    title: FastAPI
4    version: 0.1.0
5  servers:
6    - url: http://10.211.55.2:8000
7      description: Development server
8  paths:
9    "/":
10      get:
11        summary: Root
12        operationId: root__get
13        responses:
14          '200':
15            description: Successful Response
16            content:
17              application/json:
18                schema: {}
19    "/user":
20      get:
21        summary: Read User
22        operationId: read_user_user_get
23        responses:
24          '200':
25            description: Successful Response
26            content:
27              application/json:
28                schema: {}
29    "/question/{question_id}":
```

Document 1/1 paths:

Cómo



CLI

Cómo se testean – DAST :: ZAP

CLI

[Home](#)[Blog](#)[Videos](#)[Documentation](#)[Community](#)[Sponsor](#)[Download](#)

ZAP - API Scan

DOCKER > ZAP - API SCAN

ZAP - API Scan

The ZAP API scan is a script that is available in the ZAP [Docker](#) images.

It is tuned for performing scans against APIs defined by OpenAPI, SOAP, or GraphQL via either a local file or a URL.

It imports the definition that you specify and then runs an Active Scan against the URLs found. The Active Scan is tuned to APIs, so it doesn't bother looking for things like XSSs.

It also includes 2 scripts that:

- Raise alerts for any HTTP Server Error response codes
- Raise alerts for any URLs that return content types that are not usually associated with APIs

Usage

```
Usage: zap-api-scan.py -t <target> -f <format> [options]
  -t target          target API definition, OpenAPI or SOAP, local file or URL, e.g. https://www.example.com/openapi.json
  -f format          openapi, soap, or graphql
```

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

Cómo

Chrome File Edit View History Bookmarks Profiles Tab Window Help

42Crunch/apisecurity-tutorial x +

github.com/42Crunch/apisecurity-tutorial

apisecurity-tutorial Public

Watch 0 Fork 9 Star 3

main 1 Branch 0 Tags

Go to file t + <> Code

isamauny Update demo endpoint ✓ 0c345a7 · 1 hour ago 64 Commits

.42c	Use demo endpoint	1 hour ago
.github/workflows	Update workflows triggers	1 hour ago
.vscode	move scan conf out	3 weeks ago
api-specifications	Update demo endpoint	1 hour ago
graphics	V1 release update	3 months ago
.gitignore	Update .gitignore	1 hour ago
LICENSE	Initial commit	3 months ago
README.md	fix formatting	2 weeks ago
azure-pipelines-audit.yaml	Revert to INFO mode	last week
azure-pipelines-scan.yaml	Revert to INFO mode	last week
docker-compose.yaml	Tutorial V1	3 months ago

README GPL-3.0 license

42Crunch API Security Testing Tutorial

About

A sample API and OpenAPI files to test 42Crunch Freemium services

42crunch.com/free-user-faq

security scan audit apis freemium

Readme
GPL-3.0 license
Activity
Custom properties
3 stars
0 watching
9 forks
Report repository

Languages

Python 100.0%

CLI

Conclusiones

- Para un buen análisis de la API, todo empieza por **conocer su definición**.
- De lo contrario, puede que no estemos probando todos los *end-points*
- Las pruebas que se hagan serán tan buenas como bien definidas estén las API.
- Se puede (y se debe) integrar la definición **en la fase de desarrollo**

¡ Gracias !

