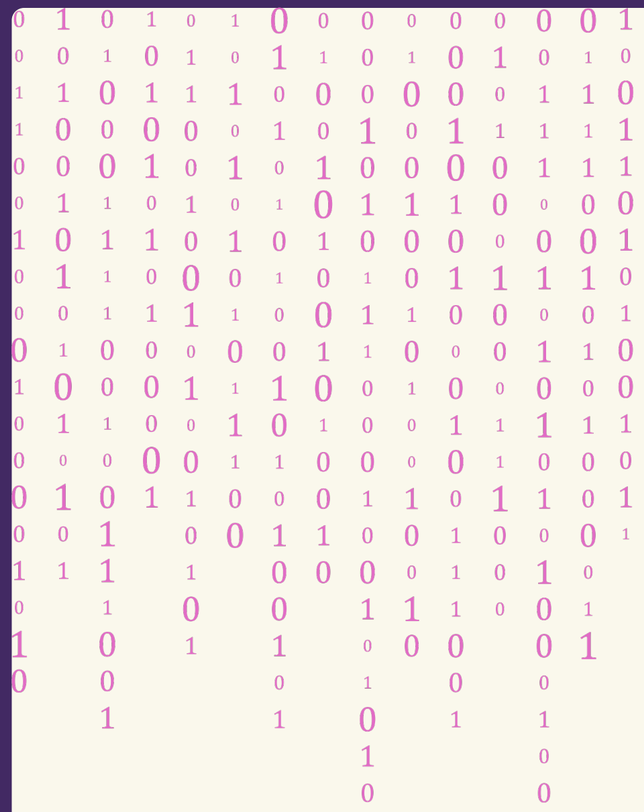

expoQA®25
MADRID
May 20th,
21st & 22nd
2025
sogeti
expoQA
expoqa.eu



Cybersecurity in Action!

Testing for a Secure Digital Future

#Masterclass #expoQA25

Sara Martínez

MAIN GOALS

SSDLC

Secure
Software
Development
Life
Cycle

PRESSURE

Be careful about new vulnerabilities, risks and attacks

AUTOMATION

Testing team goals:
Finding new security bugs

PROTECTION

Protect your products from the beginning

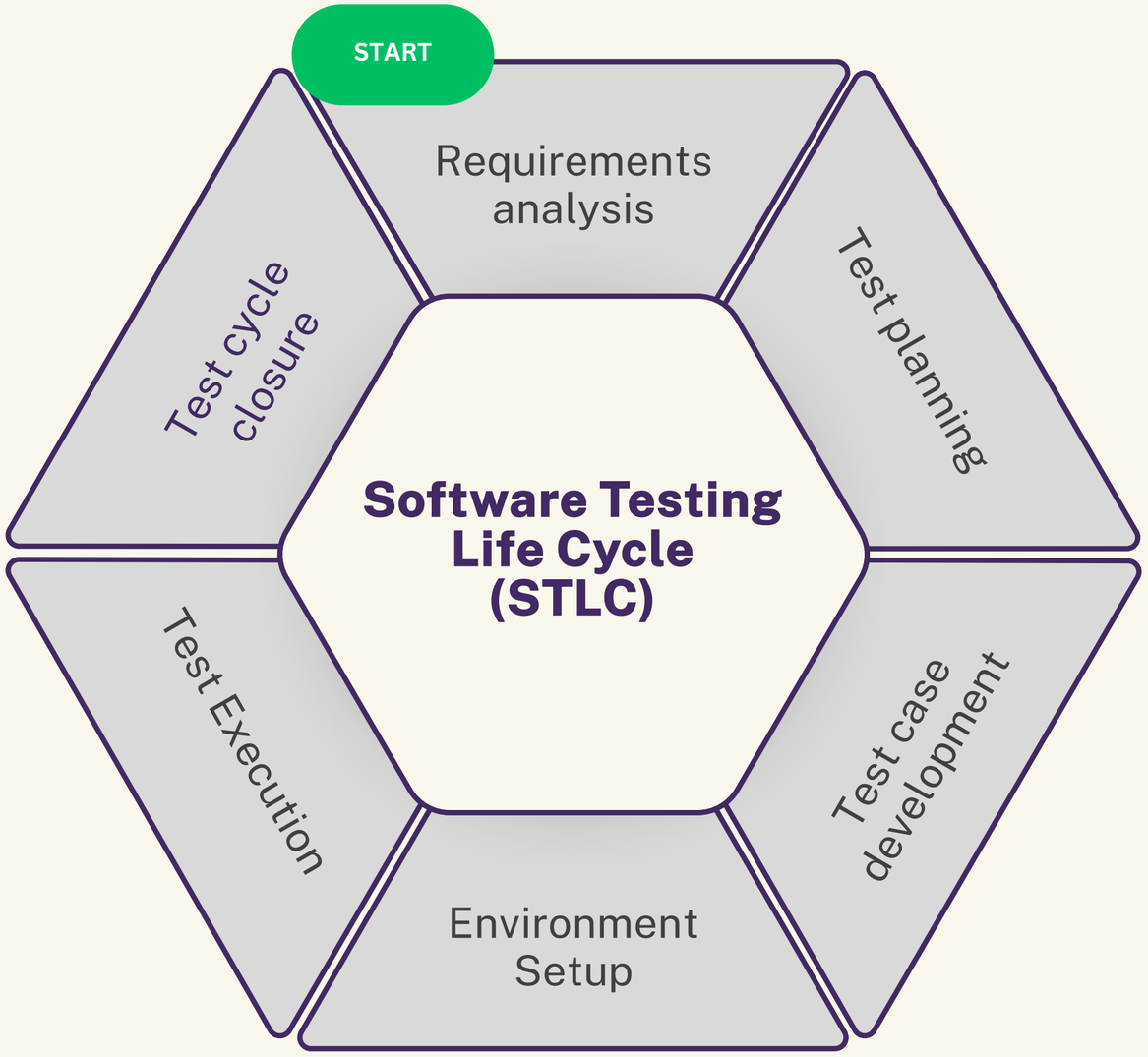
AGENDA

- CONTEXT: CODE, QUALITY, CYBERSECURITY, RISKS AND VULNERABILITIES
- QUALITY AND CYBERSECURITY CULTURE
- CODE ANALYSIS TOOLS
- HANDS ON! BREAK THE CODE
- SECURITY TEST AUTOMATION
- CI: GITHUB ACTIONS

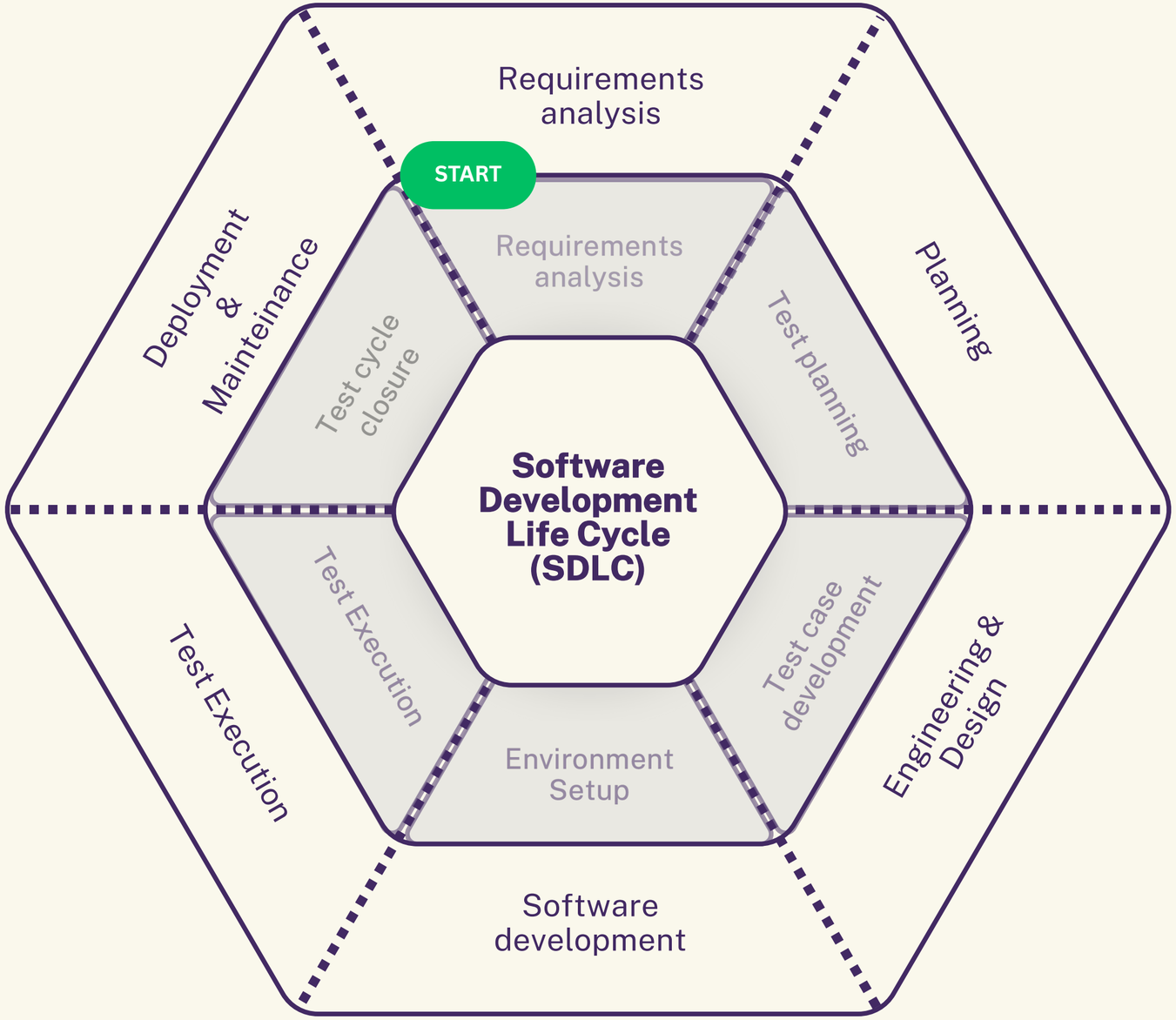
A LITTLE BIT OF CONTEXT...

....let's talk about code, quality, cybersecurity, risks and vulnerabilities

Software Testing Life Cycle



Software Development Life Cycle



CYBERSECURITY

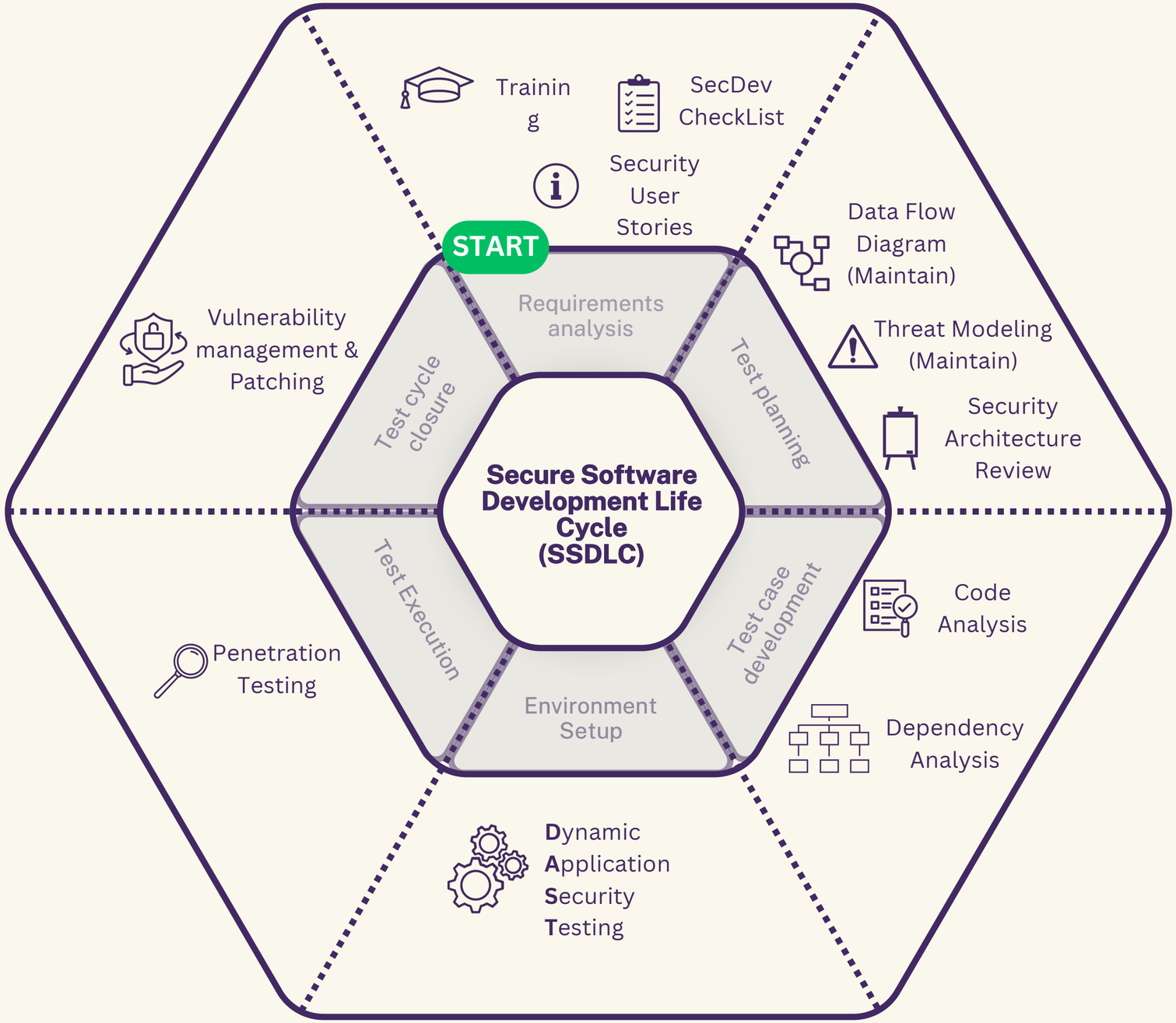
«**Cybersecurity** is the **art** of **protecting** networks, devices, and data from unauthorized access or criminal use and the practice of **ensuring** confidentiality, integrity, and availability of information.»

Source: https://www.cisa.gov/news-events/news/what-cybersecurity

SDLC Phase	STLC Phase	Security Activity
Requirements analysis	Requirements analysis	• Assess security risks and the threat landscape • Evaluate the potential impact of security incidents
Planning	Test planning	• Include security requirements • Incorporate regulatory compliance requirements
Engineering & Design	Test case development	• Develop threat models • Include security considerations in the architecture plan • Evaluate the impact of security on design decisions
Software development	Environment Setup	• Train developers on secure coding practices • Conduct static and dynamic application security testing • Evaluate software dependencies and reduce security risks
Test Execution	Test Execution	• Pentesting • Evaluate the security of the deployment environment • Review security configurations
Deployment	Test cycle closure	• Monitor the system for threats • Eliminate vulnerabilities and respond to intrusions

Source: https://www.redhat.com/es/topics/security/software-development-lifecycle-security

Secure Software Development Life Cycle



If you are part of the cycle you will have **responsibility** for **quality** and **cybersecurity**.

SOFTWARE **D**EVELOPER **E**NGINEER IN **T**EST

01. Development and testing skills.

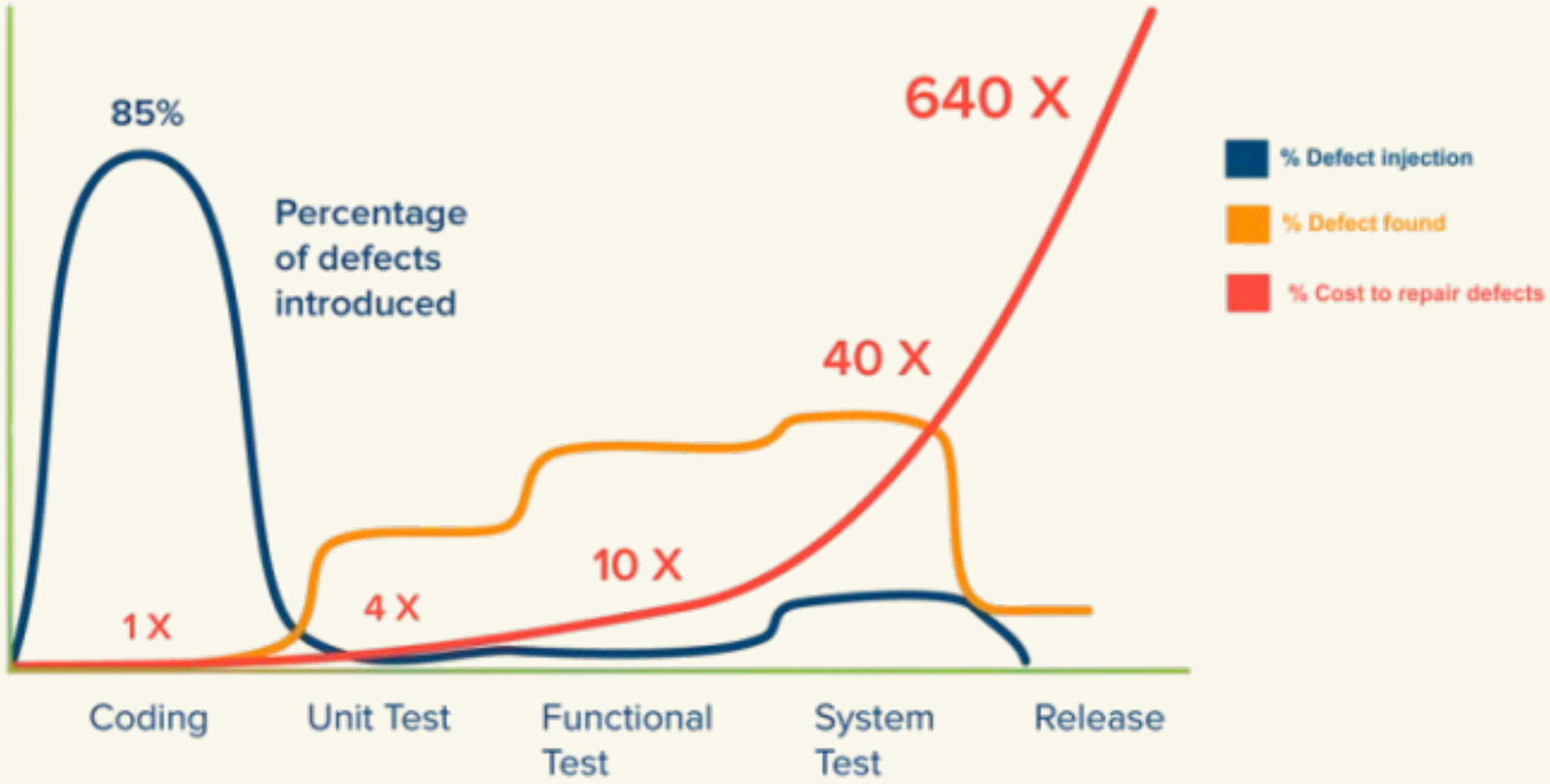
02. It takes part in the entire product life cycle.

03. Testing pyramid

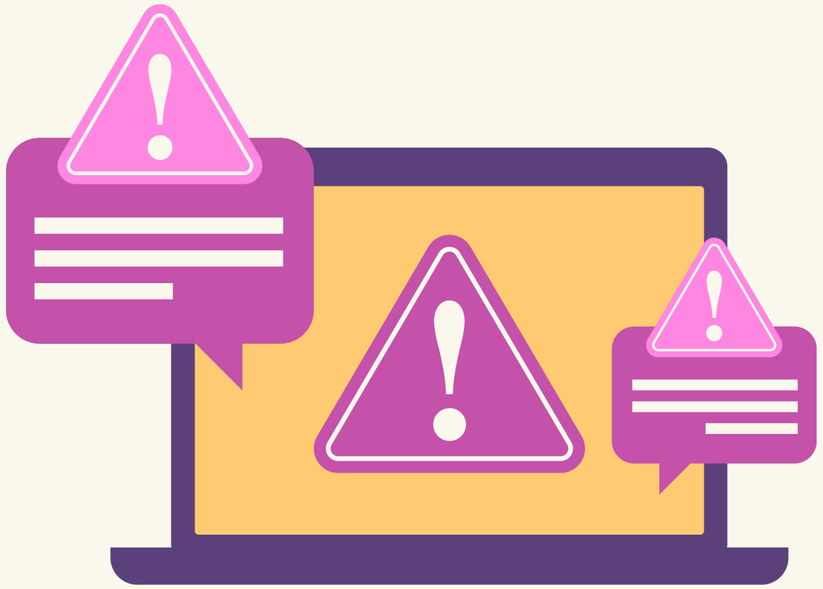
04. Knowledge focused on product stability, robustness and performance.

05. Focus on efficiency and automation.

SOFTWARE DEVELOPER ENGINEER IN TEST



WEAKNESSES
VULNERABILITIES
RISKS
THREATS



CWE

Common Weakness Enumeration (CWE) is a universal online dictionary of weaknesses that have been found in computer software.

TOP CWE OCCURRENCE

Implementation 85.71%

Architecture & Design 57.2%

Operation 17.9%

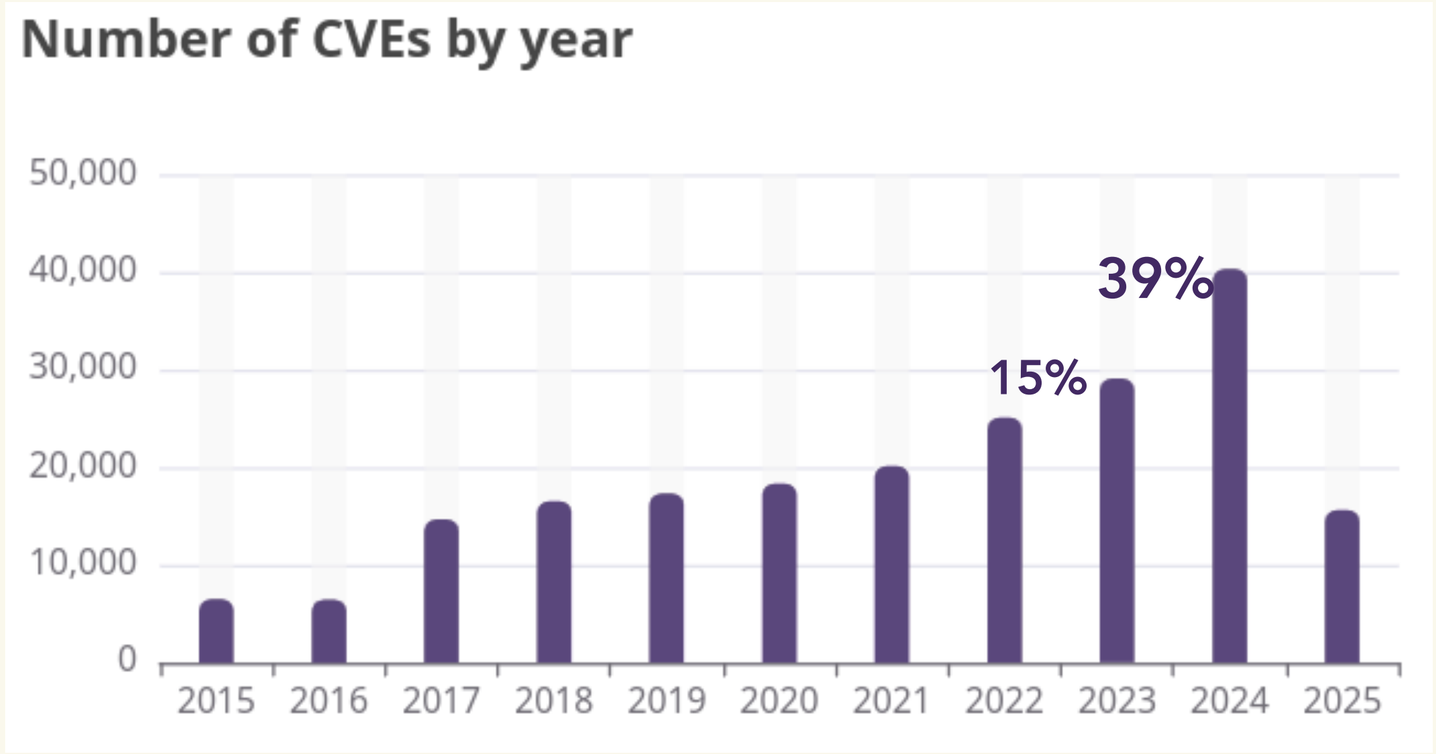
Installation 7.2%

Source: https://waverleysoftware.com/blog/top-software-vulnerabilities/

Source: https://cwe.mitre.org

CVE

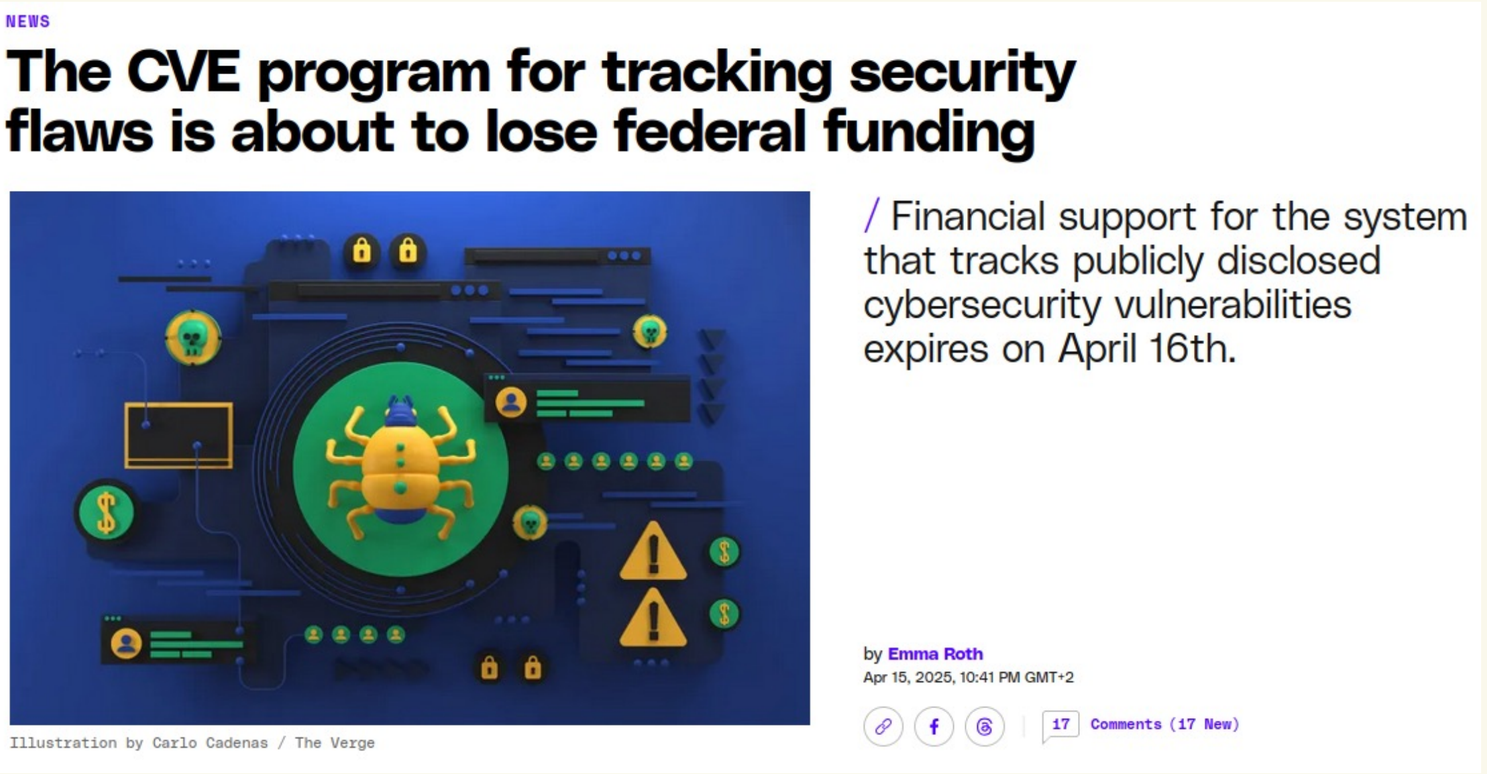
Common Vulnerabilities and Exposures (CVE) system provides a reference method for publicly known information-security vulnerabilities and exposures.



Source: https://www.cvedetails.com

CVE

Common Vulnerabilities and Exposures (CVE) system provides a reference method for publicly known information-security vulnerabilities and exposures.



NEWS

The CVE program for tracking security flaws is about to lose federal funding

/ Financial support for the system that tracks publicly disclosed cybersecurity vulnerabilities expires on April 16th.

Illustration by Carlo Cadenas / The Verge

by **Emma Roth**
Apr 15, 2025, 10:41 PM GMT+2

17 Comments (17 New)

Source: https://www.theverge.com/news/649314/cve-mitre-funding-vulnerabilities-exposures-funding


Security Vulnerabilities
Everywhere...

CrowdStrike Chaos Highlights Key Cyber Vulnerabilities with Software Updates

Veeam Software Vulnerabilities Let Attackers Trigger Remote Code Execution

Third-Party Cyber Risk Impacts the Health Care Sector the Most. Here's How to Prepare.

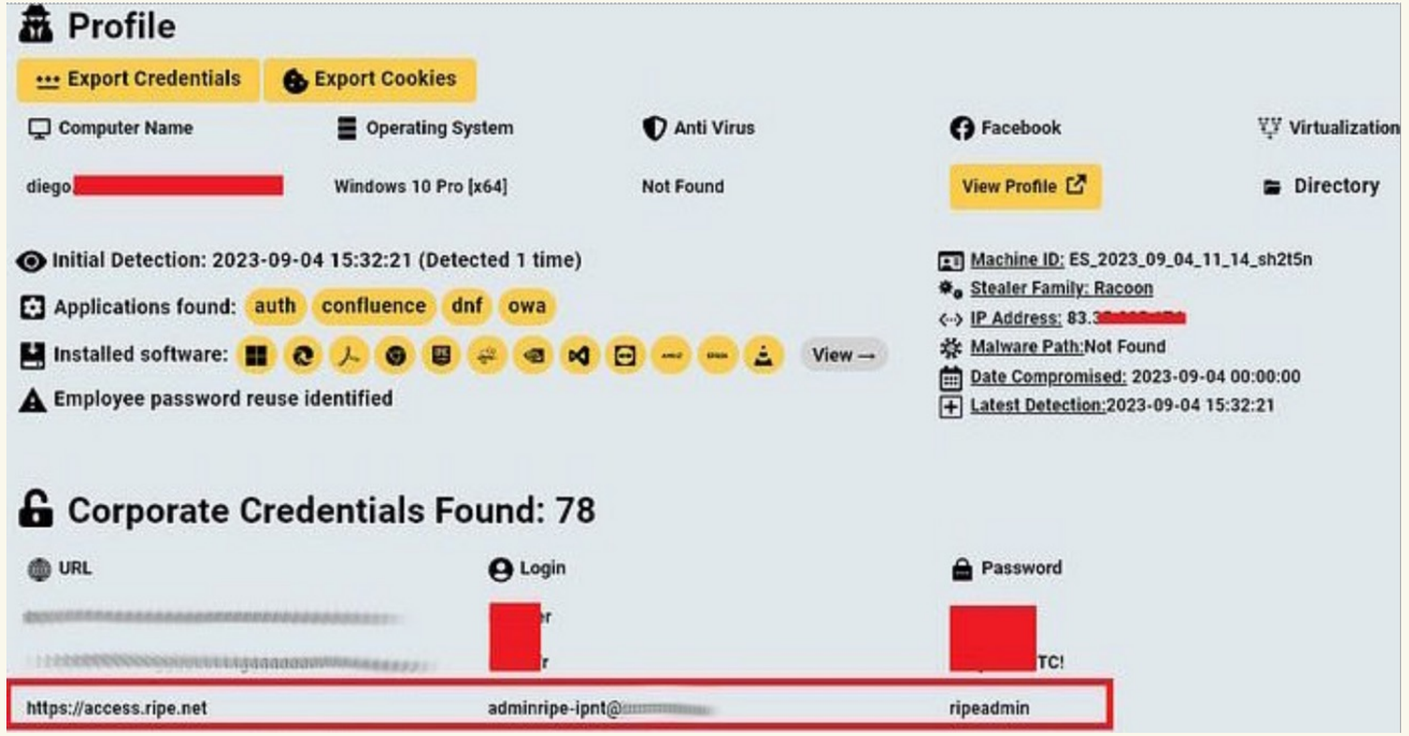
How to Secure Open Source Software: The Dilemma of the XZ Utils Backdoor

Microsoft Employees Data Exposed Via Third-Party Breach

CosmicBeetle Exploiting Old Vulnerabilities To Attack SMBs All Over The World

RIPE Account Hacking Leads to Major Internet Outage

Malware	Password exposure	Exploit	Result
Racoon	Sep23	Jan24	~1,3M users 4h service

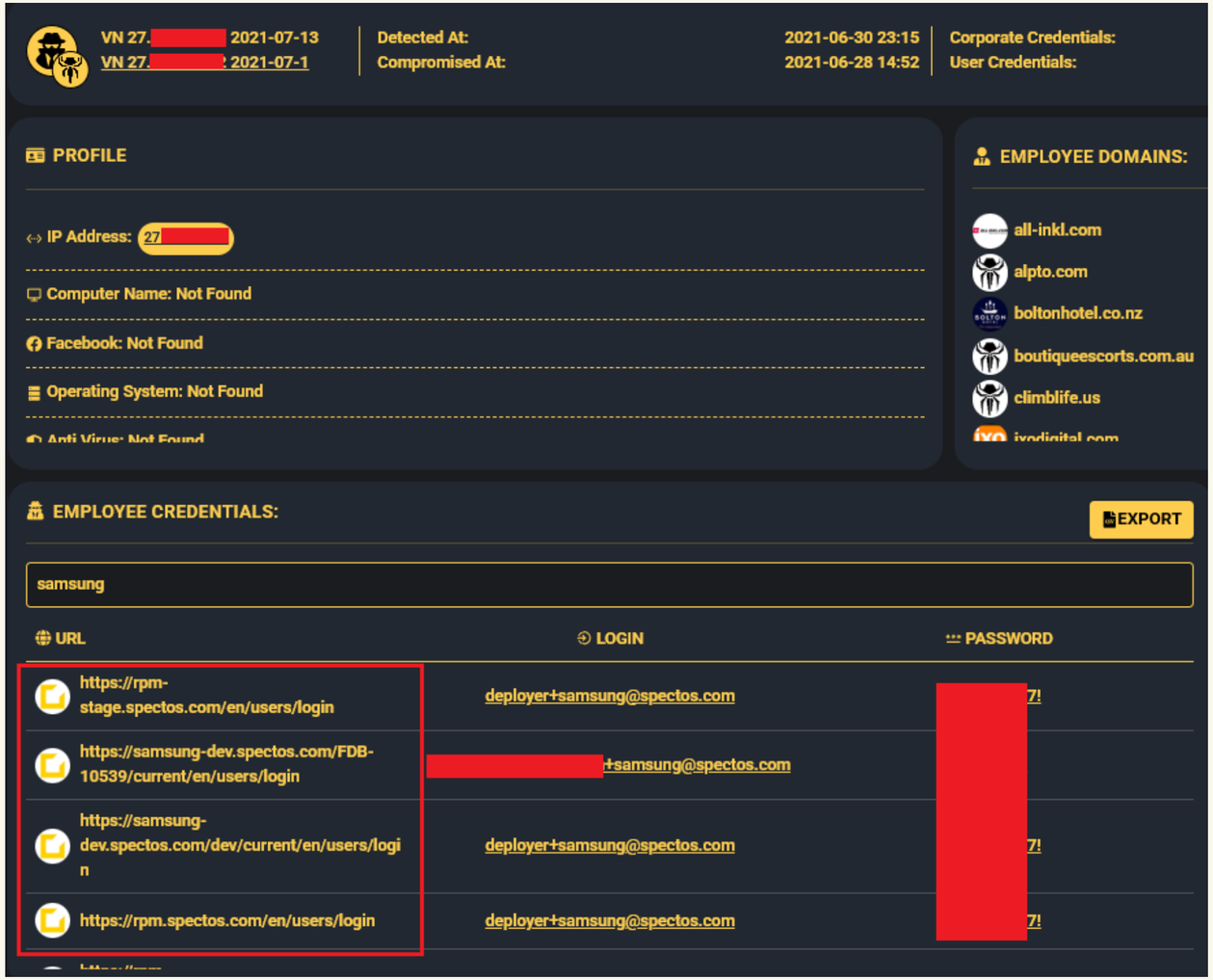


Source: https://www.securityweek.com/ripe-account-hacking-leads-to-major-internet-outage-at-orange-spain/

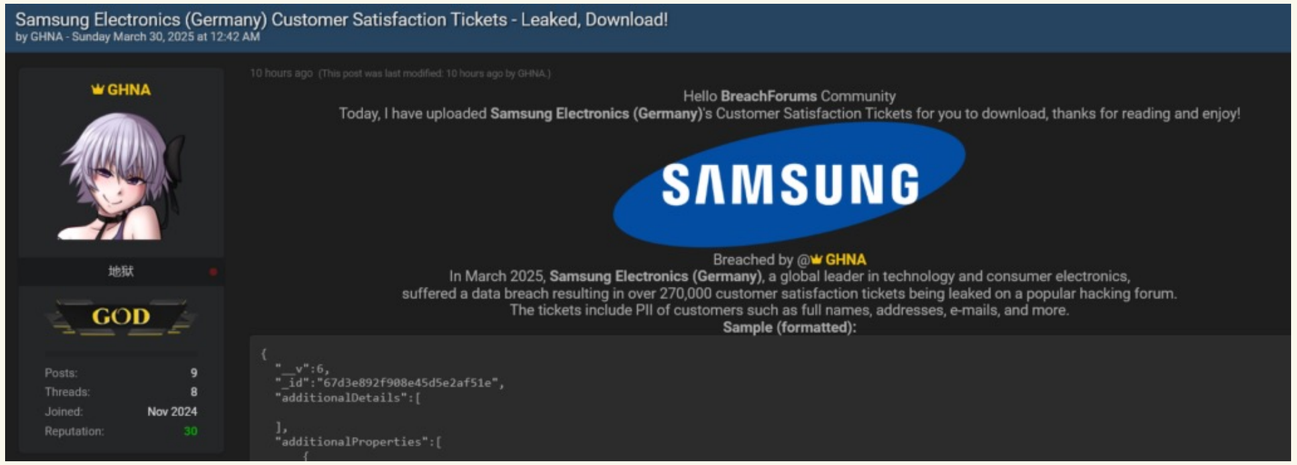
Samsung Tickets Data Leak: Infostealers Strike Again in Massive Free Dump

Infostealer	Password exposure	Exploit	Result
Racoon	2021	March 2025	270,000 customer tickets leaked

Source: https://www.infostealers.com/article/samsung-tickets-data-leak-infostealers-strike-again-in-massive-free-dump/



Samsung Tickets Data Leak: Infostealers Strike Again in Massive Free Dump



Infostealer	Password exposure	Exploit	Result
Racoon	2021	March 2025	270,000 customer tickets leaked

Source: https://www.infostealers.com/article/samsung-tickets-data-leak-infostealers-strike-again-in-massive-free-dump/

CCOO

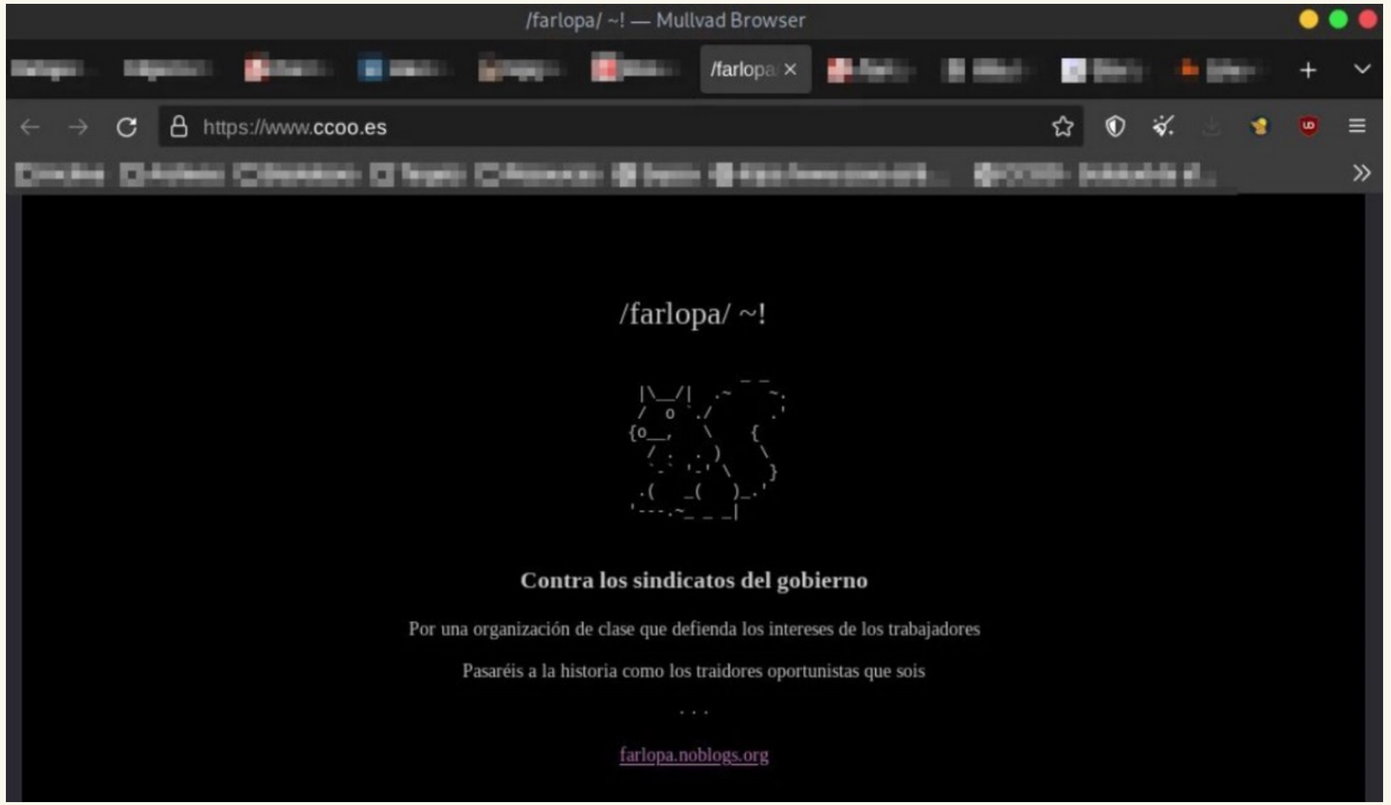
Web Inspection

E: on url

Register form

File upload (.php)

File execution



Source: https://blog.elhacker.net/2024/02/ccoo-comisiones-obreras-filtracion-22-mil-datos-personales.html

CCOO

Web Inspection

E: on url

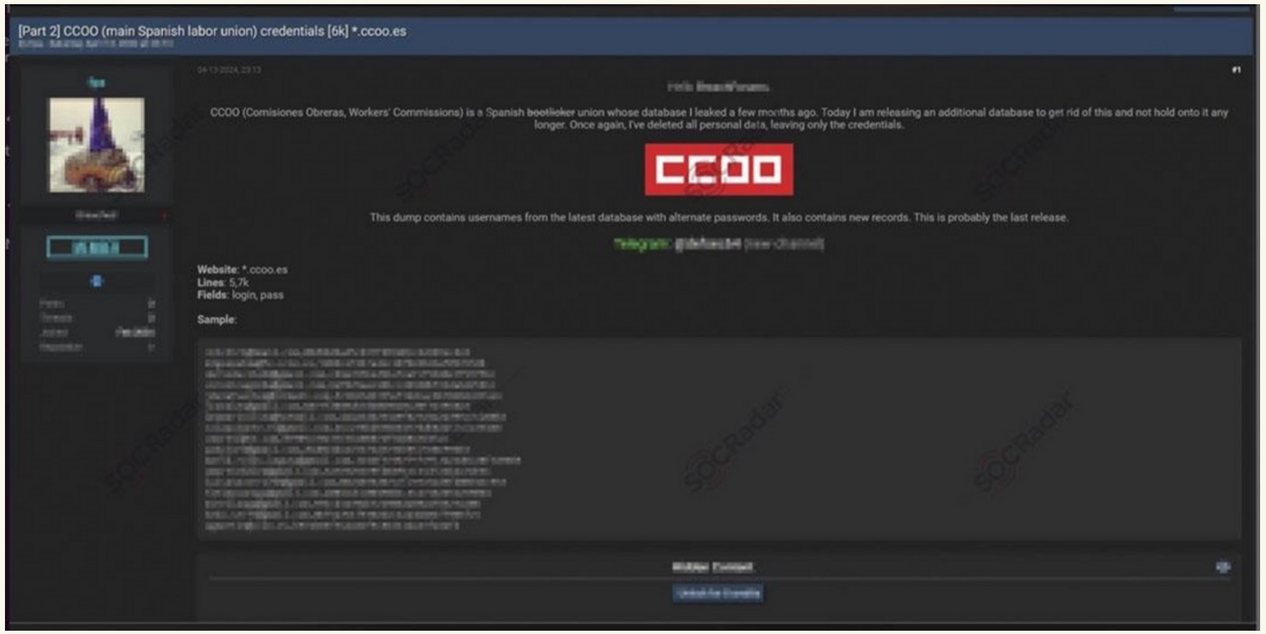
Register form

File upload (.php)

File execution

Extra attack

root access
CVE-2017-7533
>6K user credentials



Source: https://blog.elhacker.net/2024/02/ccoo-comisiones-obreras-filtracion-22-mil-datos-personales.html

QUALITY AND CYBERSECURITY CULTURE

CYBERSECURITY CULTURE

TOPICS

- Manual
- Functional
- Non Functional
- Mobile Testing
- Automation
- Management
- Accessibility
- AI
- Others...

QUALITY CULTURE

RESOURCES

- Courses
- Certifications:
  - ISTQB, CSTE, CSQA, CAST...
- Conferences:
  - expoQA, EuroSTAR, AgileTestingDays...
- Communities

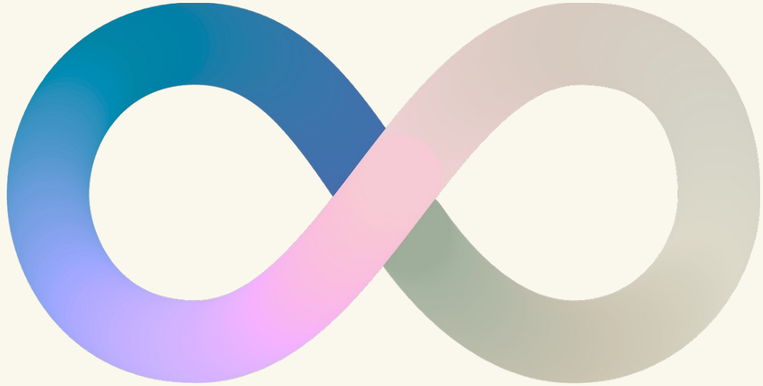
CYBERSECURITY CULTURE

TOPICS

- Network Security
- Cryptography
- **Application Security**
- Endpoint Security
- Identity and Access Management (IAM)
- Incident Response and Forensics
- Cloud Security
- Cyber Threat Intelligence (CTI)
- Governance, Risk, and Compliance (GRC)
- Penetration Testing and Ethical Hacking
- Others...

CYBERSECURITY CULTURE

RESOURCES



- Vocational training
- Career
- Masters Degree
- Certifications
  - CISSP, CEH, CISM, CISA, CCSP, CRISC, GSEC, CMA ...
- Training
- Conferences
- Workshops
- Bootcamps
- Open Resources and organisms:
  - OWASP
  - Portswigger
  - CISA

“Secure” SOFTWARE DEVELOPER ENGINEER IN TEST

ISTQB (International Software Testing Qualifications Board)
- Security Tester (2016)

Self-taught
- Courses
- Conferences
- Communities

Cybersecurity
- SSDLC Rules
- XOPS
- AppSec
- Masters Degree
- OWASP, MITRE
- Portswigger, Hack The Box...

CODE ANALYSIS TOOLS

ANALYSIS TOOLS

Application Security Testing

- SAST
- DAST

Artifact Security Management

- IMAGE SCANNING
- ARTIFACTORY

Application Security Testing

Static (SAST)

- Source code metrics
- **Aligned with security standards**
- Integration:
  - Code Editors
  - Version Control

- Tools:
  - https://owasp.org/www-community/Source_Code_Analysis_Tools#

Dynamic (DAST)

- Test Environment
- Isolated
- **Aligned with security standards**

- Tools:
  - https://www.jit.io/blog/top-dast-tools-for-2024

Alerts

Name	Risk Level	Number of Instances
SQL Injection - Oracle - Time Based	High	1
SQL Injection - SQLite	High	2
Absence of Anti-CSRF Tokens	Medium	3
CSP: Wildcard Directive	Medium	14
CSP: script-src unsafe-inline	Medium	14
CSP: style-src unsafe-inline	Medium	14
Content Security Policy (CSP) Header Not Set	Medium	2
Cross-Domain Misconfiguration	Medium	10
Missing Anti-clickjacking Header	Medium	13
Cookie No HttpOnly Flag	Low	8
Cookie without SameSite Attribute	Low	13
Cross-Domain JavaScript Source File Inclusion	Low	33
Secure Pages Include Mixed Content	Low	12
Server Leaks Information via "X-Powered-By" HTTP Response Header Field(s)	Low	18
Strict-Transport-Security Header Not Set	Low	111
Timestamp Disclosure - Unix	Low	12
X-Content-Type-Options Header Missing	Low	98
Authentication Request Identified	Informational	1
Cookie Poisoning	Informational	6
GET for POST	Informational	1
Information Disclosure - Suspicious Comments	Informational	50
Modern Web Application	Informational	15
Re-examine Cache-control Directives	Informational	10
Retrieved from Cache	Informational	4
Session Management Response Identified	Informational	8
User Agent Fuzzer	Informational	1335
User Controllable HTML Element Attribute (Potential XSS)	Informational	1

Artifact Security Management

Artifactory

- Access control
- Analysis of security and vulnerabilities
- **Aligned with security standards**
- Signature and validation

- Tools:
  - https://bluelight.co/blog/how-to-choose-a-container-registry

Image Scanner

- Artifactory
- **Aligned with security standards**
- Third Party Libraries
- Signature and validation

- Tools:
  - https://www.pingsafe.com/blog/container-scanning-tools/

Attack on DevOps: secrets leaked via malicious GitHub Action

How to respond to a compromised GitHub changed-files Action incident.

Kaspersky Team

March 18, 2025

tj-actions	Usage	When
/changed-files	>23,000 repositories	14 March 2025

Source: https://www.kaspersky.com/blog/malicious-github-action-changed-files/53179/

HANDS ON!
BREAK THE CODE

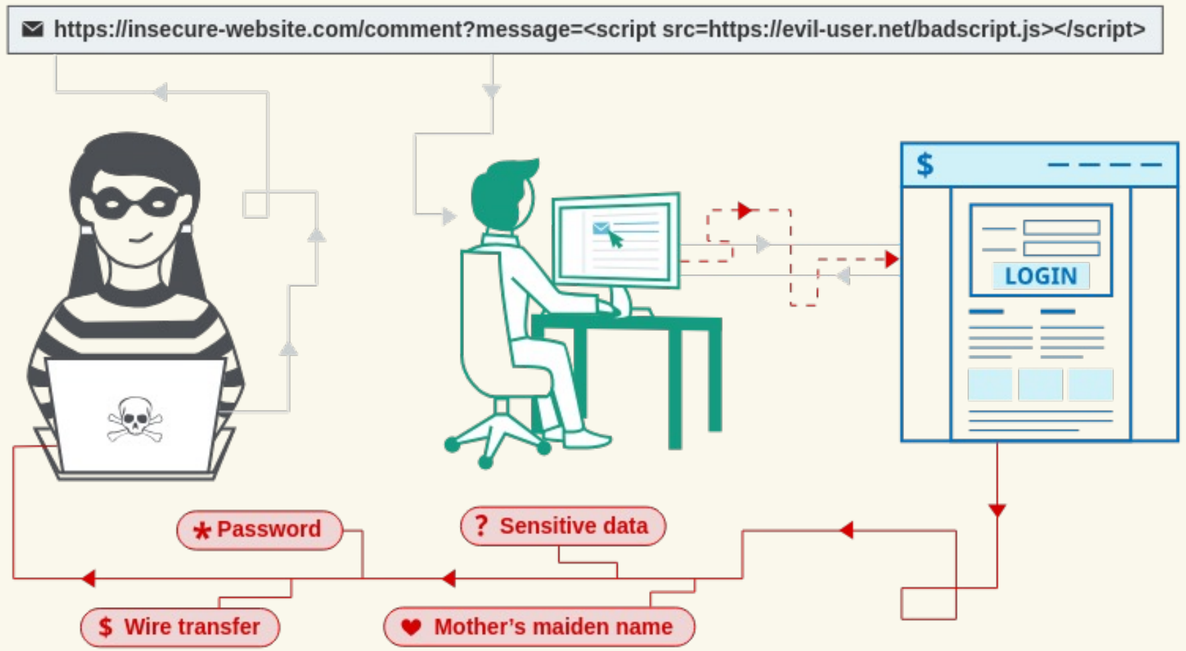
CVEs Most Exploited

Year	Overflow	Memory Corruption	Sql Injection	XSS	Directory Traversal	File Inclusion	CSRF	XXE	SSRF	Open Redirect	Input Validation
2015	1037	1104	221	776	152	6	249	50	8	46	379
2016	1180	1173	97	497	99	12	87	41	16	33	519
2017	2478	1542	505	1500	281	155	334	109	57	97	936
2018	2083	1731	503	2041	569	112	479	188	118	85	1248
2019	1205	2029	544	2387	488	126	560	137	103	121	907
2020	1217	1872	465	2201	436	110	415	119	131	100	812
2021	1663	2529	742	2724	548	91	520	126	192	133	677
2022	1863	3368	1788	3403	728	95	769	126	230	146	779
2023	1659	2236	2118	5129	764	112	1392	125	242	179	594
2024	1780	2522	2650	7456	944	257	1435	112	373	121	128
2025	369	534	644	2329	217	80	566	22	124	27	0
Total	16534	20640	10277	30443	5226	1156	6806	1155	1594	1088	6979

CVEs Most Exploited

Year	Overflow	Memory Corruption	Sql Injection	XSS	Directory Traversal	File Inclusion	CSRF	XXE	SSRF	Open Redirect	Input Validation
2015	1037	1104	221	776	152	6	249	50	8	46	379
2016	1180	1173	97	497	99	12	87	41	16	33	519
2017	2478	1542	505	1500	281	155	334	109	57	97	936
2018	2083	1731	503	2041	569	112	479	188	118	85	1248
2019	1205	2029	544	2387	488	126	560	137	103	121	907
2020	1217	1872	465	2201	436	110	415	119	131	100	812
2021	1663	2529	742	2724	548	91	520	126	192	133	677
2022	1863	3368	1788	3403	728	95	769	126	230	146	779
2023	1659	2236	2118	5129	764	112	1392	125	242	179	594
2024	1780	2522	2650	7456	944	257	1435	112	373	121	128
2025	369	534	644	2329	217	80	566	22	124	27	0
Total	16534	20640	10277	30443	5226	1156	6806	1155	1594	1088	6979

XSS - CROSS-SITE SCRIPTING

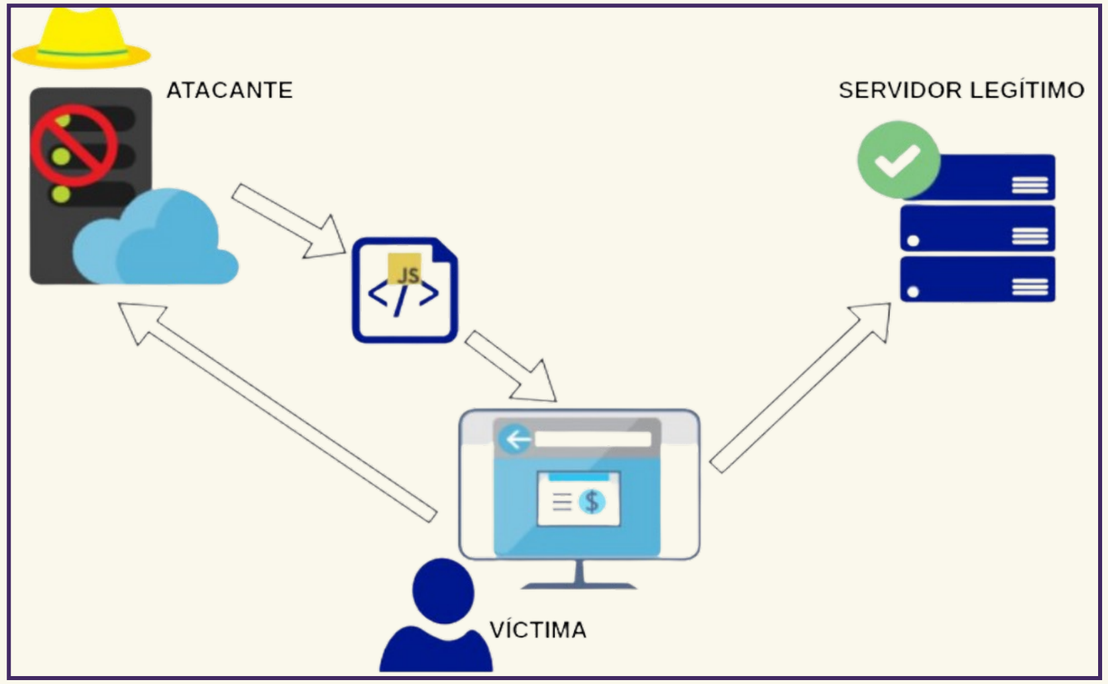


- **Reflected XSS**: malicious script comes from the current HTTP request.
- **Stored XSS**: malicious script comes from the website's database.
- **DOM-based XSS**: the vulnerability exists in client-side code rather than server-side code.

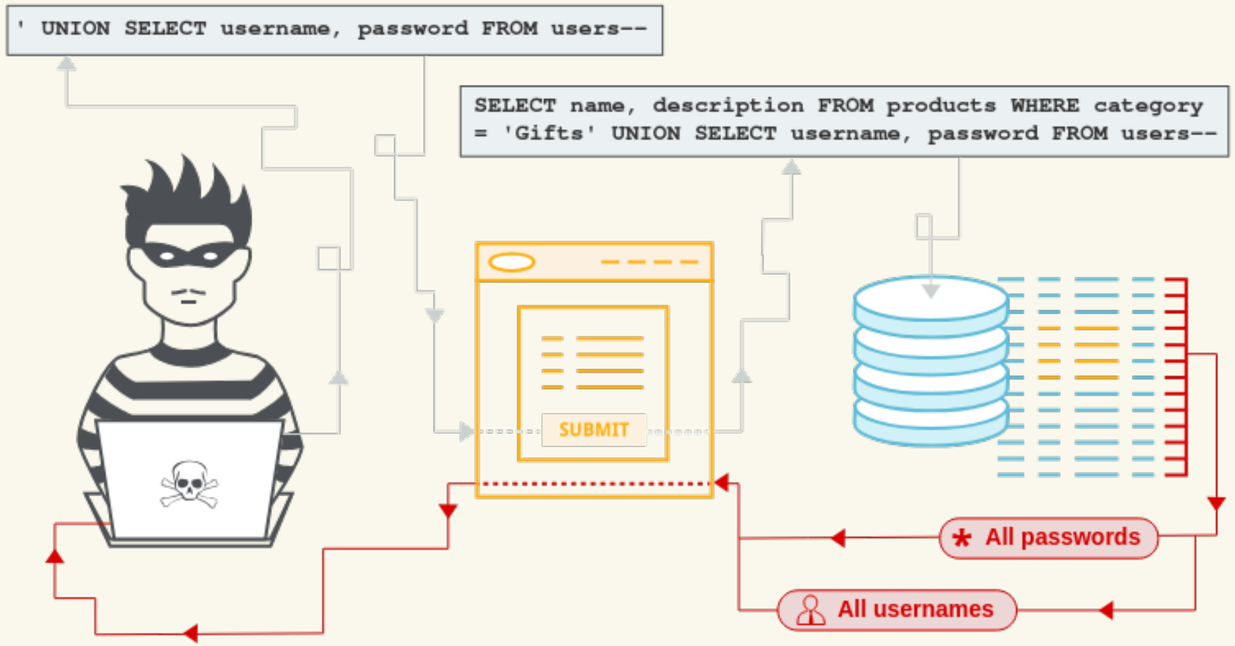
MAGECART - FORMJACKING

Irene Cotillas Torres
@IreCoT

Juan Carlos Fernández
@fernandez_jc



SQLi - SQL Injection



A successful SQL injection attack can result in unauthorized access to sensitive data, such as:

- **Passwords.**
- **Credit card details.**
- **Personal user information.**

Shopware Security Plugin Exposes Systems to SQL Injection Attacks

```
class PatchedAggregationParser extends AggregationParser
{
    public function buildAggregations(EntityDefinition
$definition, array $payload, Criteria $criteria,
SearchRequestException $searchRequestException): void
    {
        parent::buildAggregations($definition, $payload,
$criteria, $searchRequestException);
        foreach ($criteria->getAggregations() as $i =>
$aggregation) {
            if (str_contains($aggregation->getName(), '?') ||
str_contains($aggregation->getName(), ':')) {
                $searchRequestException->add(new
InvalidAggregationQueryException('Invalid character in
aggregation name'));
            }
        }
    }
}
```

Source: https://cybersecuritynews.com/shopware-security-plugin-exposes-systems/

LET'S HACK!

LAB XSS:

- https://portswigger.net/web-security/cross-site-scripting

LAB SQLi:

- https://portswigger.net/web-security/sql-injection

SECURITY TEST AUTOMATION

OWASP
Open Web Application Security Project

TOP 10:2024

01. Broken Access Control
02. Cryptographic Failures
03. Injection
04. Insecure Design
05. Security Misconfiguration
06. Vulnerable and Outdated Components
07. Identification and Authentication Failures
08. Software and Data Integrity Failures
09. Security Logging and Monitoring Failures
10. Server-Side Request Forgery (SSRF)

SECURITY TEST SCENARIOS

Roles and permisions

- Privilege scalation

Files

- Alow only valid files

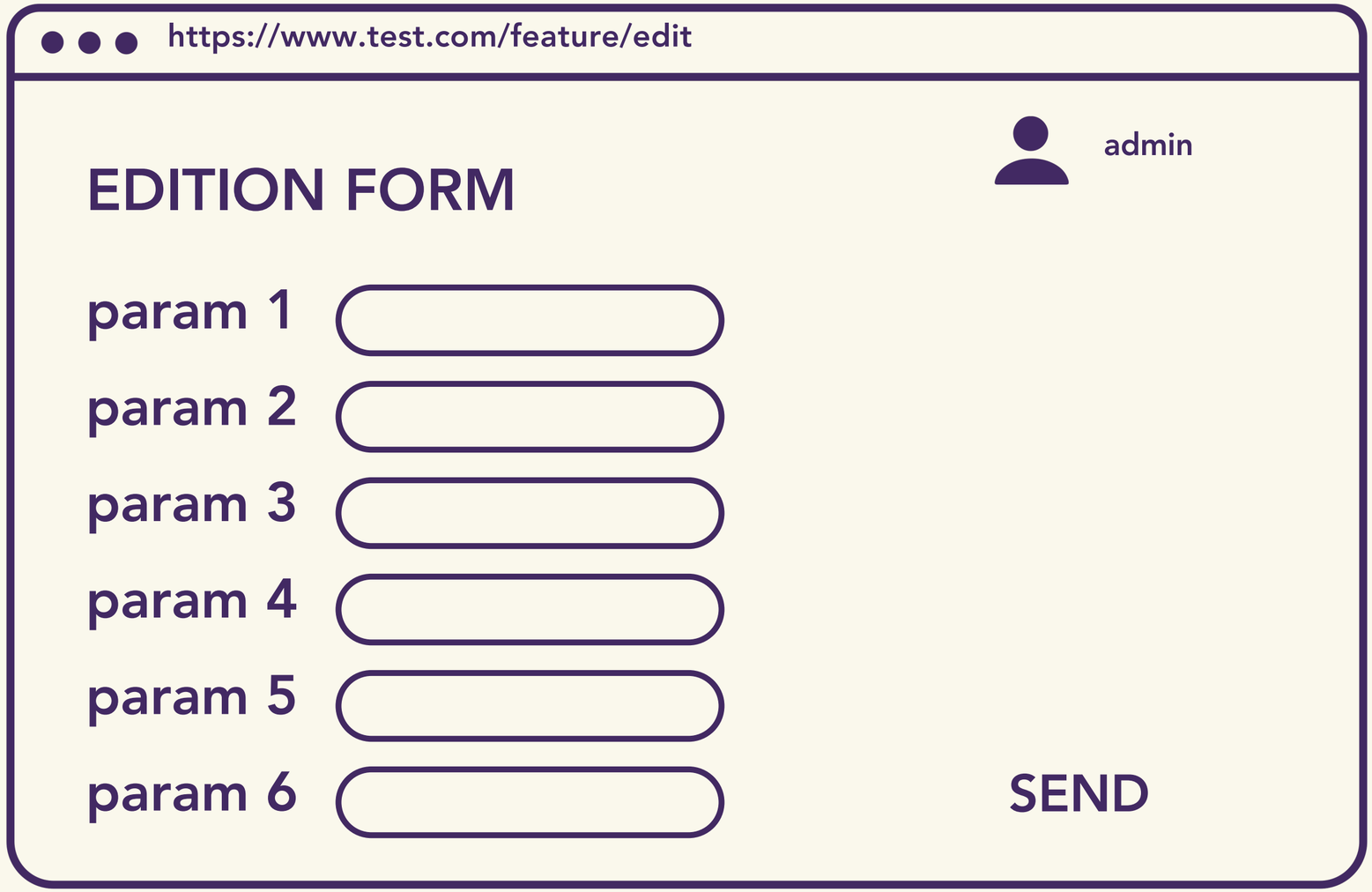
Login

- Strong Password
- 2FA
- Reestablishment procedures
- Cookies expiration

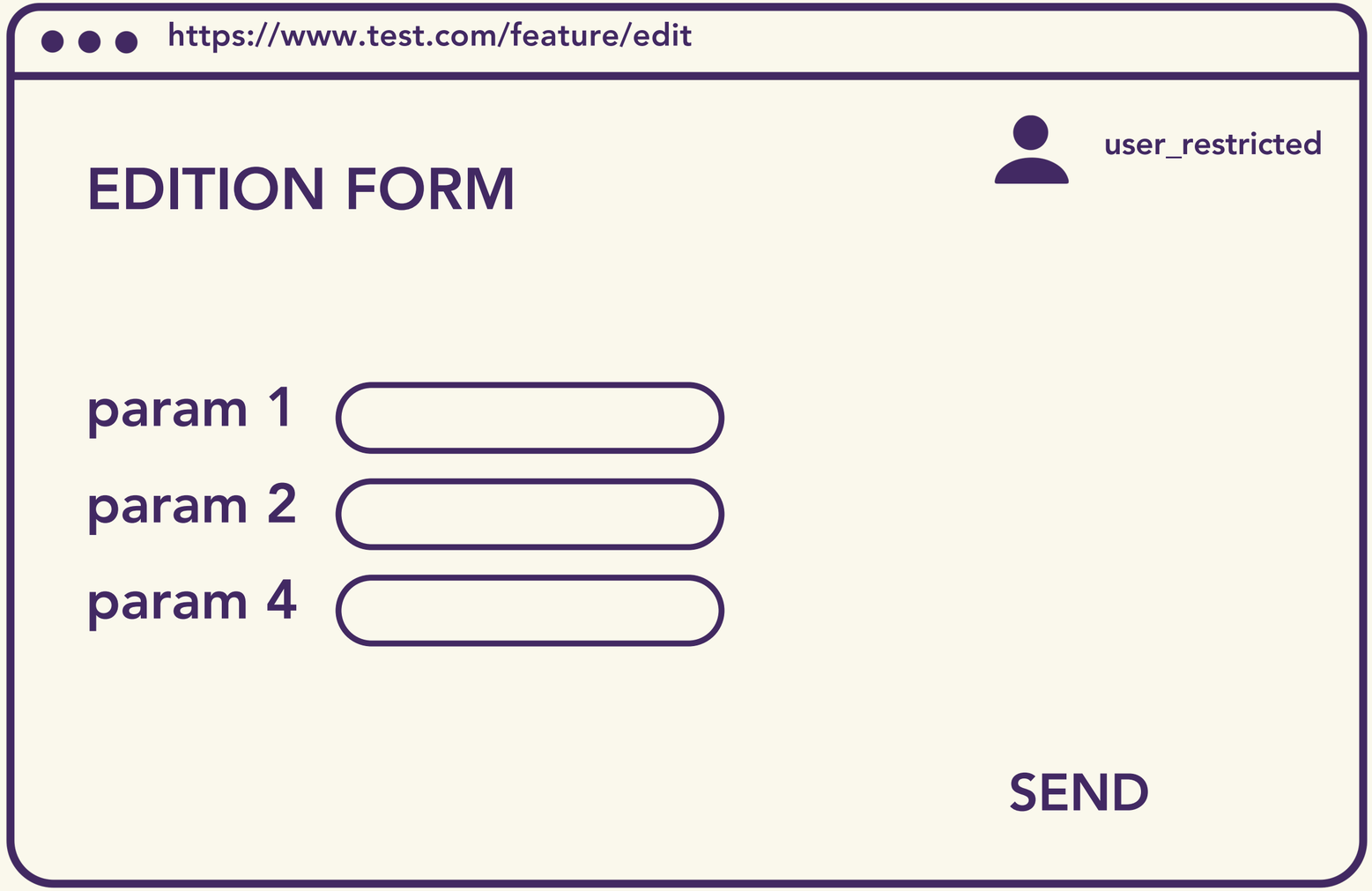
Forms

- Limit inputs (ie.XSS)

EXAMPLE: Form with admin privileges



EXAMPLE: Form with restricted privileges



WHY SO FUNCTIONAL?

```
@component
Feature: Edition form

  @dev @security
  Scenario: Check edition params for restricted user
    Given I login with user "user_with_restricted_permission"
    And I navigate to the "https://www.test.com/feature/" URL
    When I go to edition form
    Then the following params are visible and writeable
      | param 1 |
      | param 2 |
      | param 4 |
    And the following params are not visible in form
      | param 3 |
      | param 5 |
      | param 6 |
```

WHY SO FUNCTIONAL?

```
@component
Feature: Feature data visibility

  @dev @security
  Scenario: Check feature params for restricted user - API
    Given I login to host "https://www.test.com/"
      | authentication                    |
      | user_with_restricted_permission |
    When I send a "GET" request to the "feature/" api endpoint
    Then the response status code is "200"
    And the response body contains the following params
      | param 1 |
      | param 2 |
      | param 4 |
    And the response body does not contain the following params
      | param 3 |
      | param 5 |
      | param 6 |
```

WHY SO FUNCTIONAL?

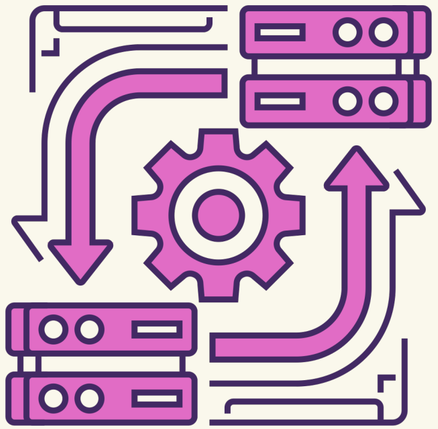
```
@dev @security
Scenario Outline: Check writable feature params for restricted user - API
  Given I login to host "https://www.test.com/"
    | authentication                    |
    | user_with_restricted_permission |
  And the body defined by file "restricted_edit_feature.json" with values
    | parameter   | value   |
    | <parameter> | <value> |
  When I send a "PUT" request to the "feature/edit" api endpoint
  Then the response status code is "200"
  Examples:
    | parameter | value |
    | param 1   | val1  |
    | param 2   | val2  |
    | param 4   | val3  |
```

WHY SO FUNCTIONAL?

```
@dev @security
Scenario Outline: Check non writable feature params for restricted user - API
  Given I login to host "https://www.test.com/"
    | authentication                    |
    | user_with_restricted_permission |
  And the body defined by file "restricted_edit_feature.json" adding values
    | parameter   | value   |
    | <parameter> | <value> |
  When I send a "PUT" request to the "feature/edit" api endpoint
  Then the response status code is "400"
  Examples:
    | parameter | value |
    | param 3   | val3  |
    | param 5   | val5  |
    | param 6   | val6  |
```

HANDS ON!

TEST AUTOMATION



https://github.com/testingsoul/cyber_automate

Toolium
- Python
- Selenium
- API
- Behave

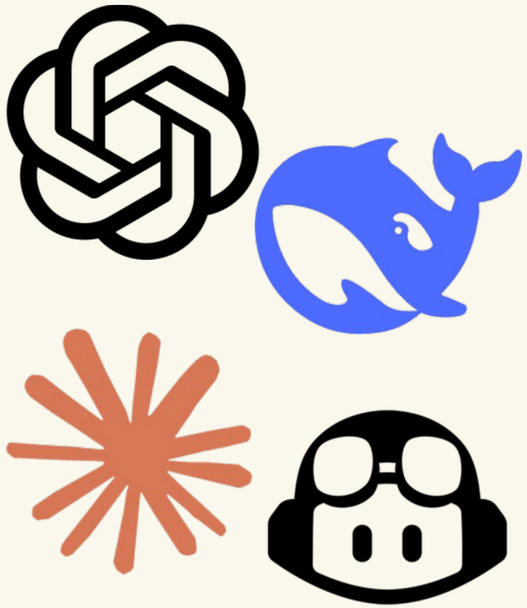
Code Security
- git-crypt / secrets
- Dependabot
- SAST
- PRs Management

```
Feature: XSS through inputs

Background:
  Given the user logs into PortSwigger

  @security
  Scenario Outline: Force XSS through input
   Given I open XSS laboratory
    When I search the value "<value>"
    Then there are no alerts in browser

    Examples:
      | value                                    |
      | <script>alert('Crash WEB')</script> |
```



USE AI TO DEFINE SECURITY TEST SCENARIOS

Roles and permisions

- Privilege scalation

Files

- Alow only valid files

Login

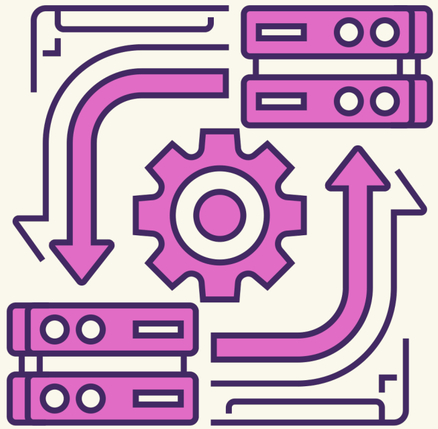
- Strong Password
- 2FA
- Reestablishment procedures
- Cookies expiration

Forms

- Limit inputs (ie.XSS)

CI: GITHUB ACTIONS

GOALS



- PULL REQUEST INTEGRATION
- EARLY FEEDBACK
- EASY REPORT

GITHUB ACTIONS STRUCTURE

```yaml
name: Test Execution

on: ...

jobs:
  build-and-test:
    name: Build and Test
    runs-on: ubuntu-latest
    strategy: ...

    steps:
    - uses: actions/checkout@v2
    - name: Set up Python ...
    - name: Unlock secrets ...
    - name: Install toolium ...
    - name: Start selenoid ...
    - name: Run acceptance tests ...
    - name: Upload Acceptance Test Results ...
    - name: Run security tests ...
    - name: Upload Security Test Results ...

  publish-acceptance-test-results: ...

  publish-security-test-results: ...
```

GITHUB ACTIONS AS SIMPLE AS...

```yaml
    - name: Run acceptance tests
      run: |
        behave test --junit --junit-directory test/output -t "~@security"
    - name: Upload Acceptance Test Results ⋯
    - name: Run security tests
      run: |
        behave test --junit --junit-directory test/output -t "@security"
    - name: Upload Security Test Results ⋯
```

Build and Test
`Process completed with exit code 1.`

⚠ **Force XSS through input -- @1.1 (features.inputs.XSS through inputs) failed:** features.inputs.XSS through inputs#L0
`security-tests-results/TESTS-features.inputs.xml [took 9s]`

⚠ **Force SQLi through url (features.url.SQLi through url) failed:** features.url.SQLi through url#L0
`security-tests-results/TESTS-features.url.xml [took 8s]`

Publish Acceptance Tests Results summary ⋯

Acceptance Tests Results

3 tests ±0 1 ✅ ±0 10s ⏱ -15s
2 suites ±0 2 💤 ±0
2 files ±0 0 ❌ ±0

Results for commit 62647455. ± Comparison against earlier commit 56d2e37a.

Job summary generated at run-time

Publish Security Tests Results summary ⋯

Security Tests Results

3 tests ±0 0 ✅ -1 17s ⏱ -8s
2 suites ±0 1 💤 -1
2 files ±0 2 ❌ +2

For more details on these failures, see this check.

Results for commit 62647455. ± Comparison against earlier commit 56d2e37a.

Job summary generated at run-time

CONCLUSIONS

CONCLUSIONS

01. Security is a Continuous Process

02. Shift Left for Stronger Security

03. Raising of vulnerabilities and risks

04. Awareness and Culture Matter

05. Security testing is also functional

THANKS!

ANY QUESTION?

@swtestingsoul

saramartinezgine
r

sara@testingsoul.com

www.testingsoul.com


expoQA25
MADRID
May 20th,
21st & 22nd
2025
Thank you for attending
expoqa.eu