(In)Secure Bank

Gregor Spagnolo

OWASP Italy Day 2023

Politecnico of Milan - 11th September 2023

About me

Gregor Spagnolo

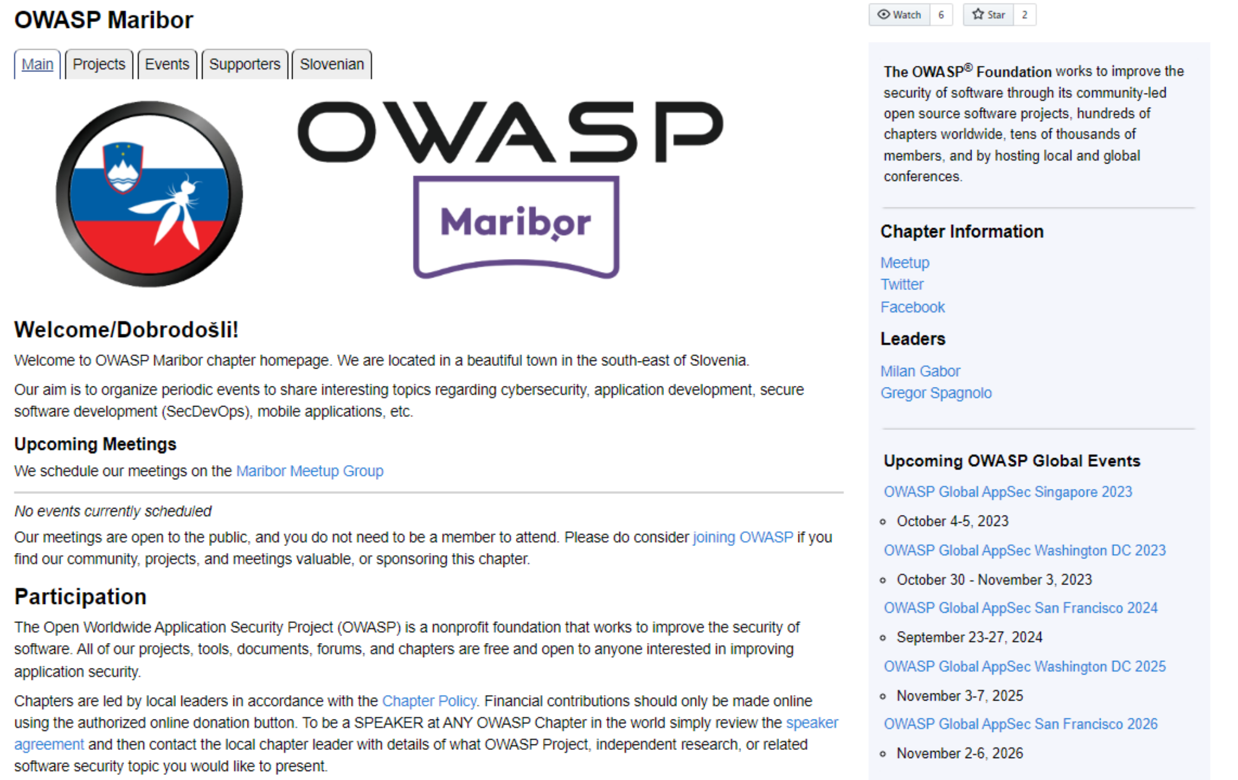
- + 15 years of dev experience
- Trainer
- Security engineer
- SSRD d.o.o. owner
- Like to have pfun

OWASP Maribor chapter co-lead

@gregorspangolo

gregorspangolo

OWASP Maribor



OWASP Maribor

Welcome/Dobrodošli!

Welcome to OWASP Maribor chapter homepage. We are located in a beautiful town in the south-east of Slovenia.

Our aim is to organize periodic events to share interesting topics regarding cybersecurity, application development, secure software development (SecDevOps), mobile applications, etc.

Upcoming Meetings

We schedule our meetings on the Maribor Meetup Group

No events currently scheduled

Our meetings are open to the public, and you do not need to be a member to attend. Please do consider joining OWASP if you find our community, projects, and meetings valuable, or sponsoring this chapter.

Participation

The Open Worldwide Application Security Project (OWASP) is a nonprofit foundation that works to improve the security of software. All of our projects, tools, documents, forums, and chapters are free and open to anyone interested in improving application security.

Chapters are led by local leaders in accordance with the Chapter Policy. Financial contributions should only be made online using the authorized online donation button. To be a SPEAKER at ANY OWASP Chapter in the world simply review the speaker agreement and then contact the local chapter leader with details of what OWASP Project, independent research, or related software security topic you would like to present.

The OWASP® Foundation works to improve the security of software through its community-led open source software projects, hundreds of chapters worldwide, tens of thousands of members, and by hosting local and global conferences.

Chapter Information

Meetup
Twitter
Facebook

Leaders

Milan Gabor
Gregor Spagnolo

Upcoming OWASP Global Events

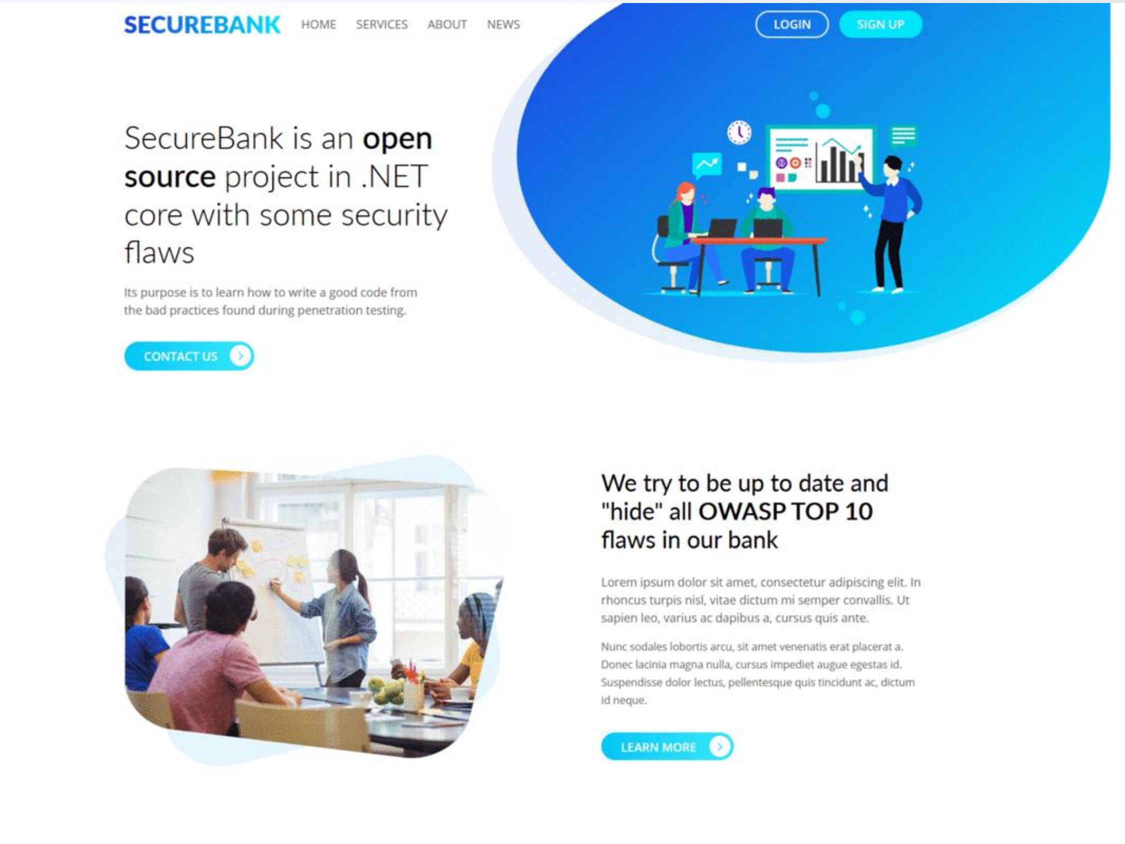
OWASP Global AppSec Singapore 2023
- October 4-5, 2023

OWASP Global AppSec Washington DC 2023
- October 30 - November 3, 2023

OWASP Global AppSec San Francisco 2024
- September 23-27, 2024

OWASP Global AppSec Washington DC 2025
- November 3-7, 2025

OWASP Global AppSec San Francisco 2026
- November 2-6, 2026



SecureBank is an **open source** project in .NET core with some security flaws

Its purpose is to learn how to write a good code from the bad practices found during penetration testing.

We try to be up to date and "hide" all **OWASP TOP 10** flaws in our bank

Lorem ipsum dolor sit amet, consectetur adipiscing elit. In rhoncus turpis nisl, vitae dictum mi semper convallis. Ut sapien leo, varius ac dapibus a, cursus quis ante.

Nunc sodales lobortis arcu, sit amet venenatis erat placerat a. Donec lacinia magna nulla, cursus impediet augue egestas id. Suspendisse dolor lectus, pellentesque quis tincidunt ac, dictum id neque.

Developers

- Build
- Create

- Hackers
  - Kill dreams

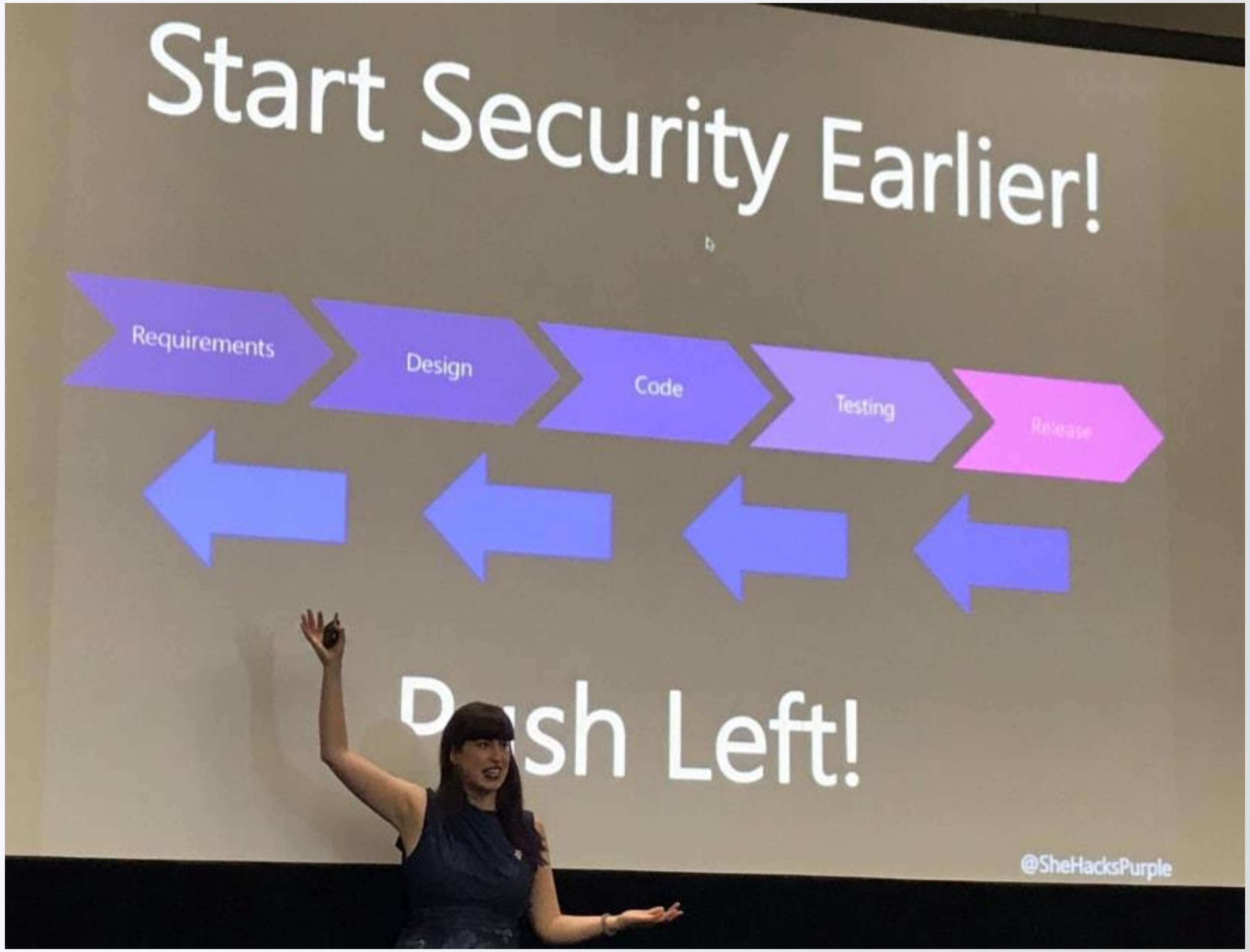
Developers

- Build
- Create

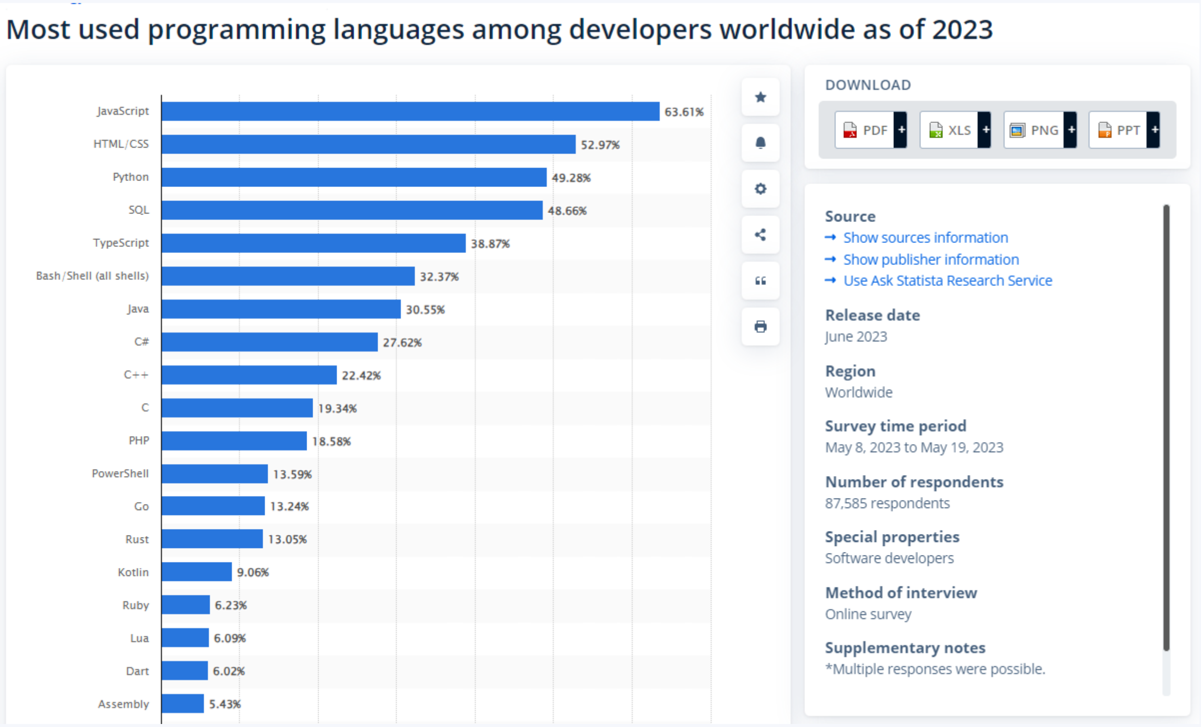
- Hackers
  - Kill dreams

Pushing security to the left

- SDLC (Jan 2002)
  - 1998 (SQLi)
  - 2021 lost first place
- Training
- Architecture design
- CI/CD integration
- Vulnerability management



Programming languages



Most used programming languages among developers worldwide as of 2023

Language	Share
JavaScript	63.61%
HTML/CSS	52.97%
Python	49.28%
SQL	48.66%
TypeScript	38.87%
Bash/Shell (all shells)	32.37%
Java	30.55%
C#	27.62%
C++	22.42%
C	19.34%
PHP	18.58%
PowerShell	13.59%
Go	13.24%
Rust	13.05%
Kotlin	9.06%
Ruby	6.23%
Lua	6.09%
Dart	6.02%
Assembly	5.43%

Source
→ Show sources information
→ Show publisher information
→ Use Ask Statista Research Service

Release date
June 2023

Region
Worldwide

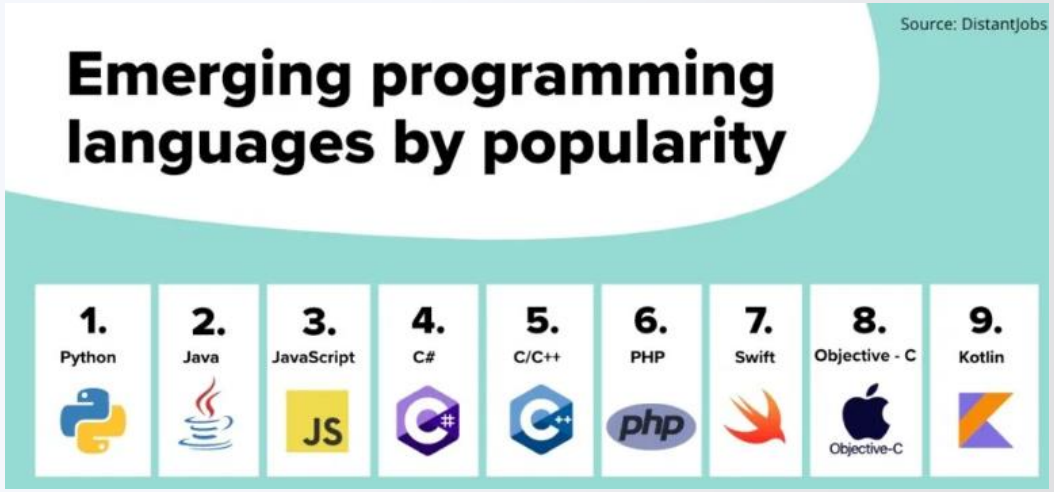
Survey time period
May 8, 2023 to May 19, 2023

Number of respondents
87,585 respondents

Special properties
Software developers

Method of interview
Online survey

Supplementary notes
*Multiple responses were possible.



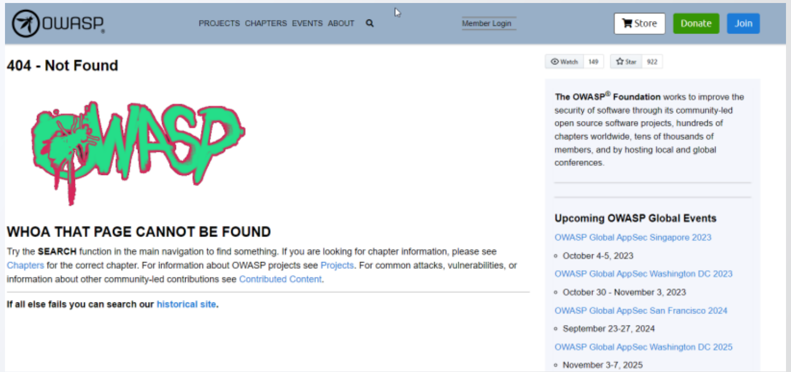
Source: DistantJobs

Emerging programming languages by popularity

1. Python
2. Java
3. JavaScript
4. C#
5. C/C++
6. PHP
7. Swift
8. Objective - C
9. Kotlin

.NET core

- https://cheatsheetseries.owasp.org/cheatsheets/DotNet_Security_Cheat_Sheet.html
- Projects:
    - https://owasp.org/www-community/vulnerabilities/OWASP_NET_Vulnerability_Research



404 - Not Found

WHOA THAT PAGE CANNOT BE FOUND

Try the **SEARCH** function in the main navigation to find something. If you are looking for chapter information, please see Chapters for the correct chapter. For information about OWASP projects see Projects. For common attacks, vulnerabilities, or information about other community-led contributions see Contributed Content.

If all else fails you can search our historical site.

The OWASP® Foundation works to improve the security of software through its community-led open source software projects, hundreds of chapters worldwide, tens of thousands of members, and by hosting local and global conferences.

Upcoming OWASP Global Events

OWASP Global AppSec Singapore 2023
- October 4-5, 2023

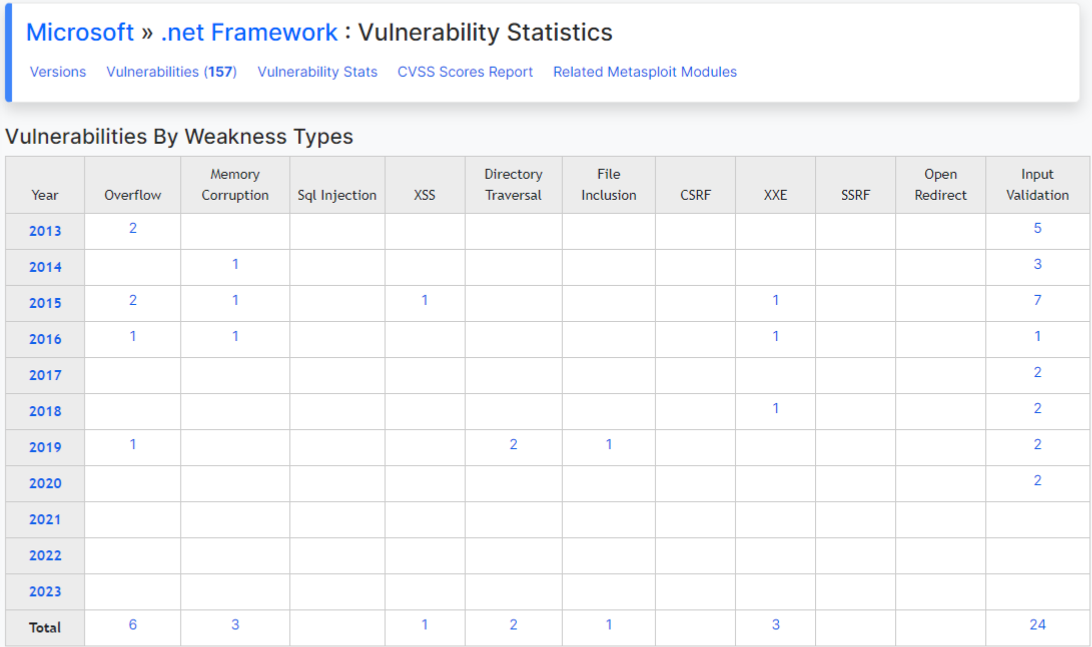
OWASP Global AppSec Washington DC 2023
- October 30 - November 3, 2023

OWASP Global AppSec San Francisco 2024
- September 23-27, 2024

OWASP Global AppSec Washington DC 2025
- November 3-7, 2025

.NET

- Secure by default

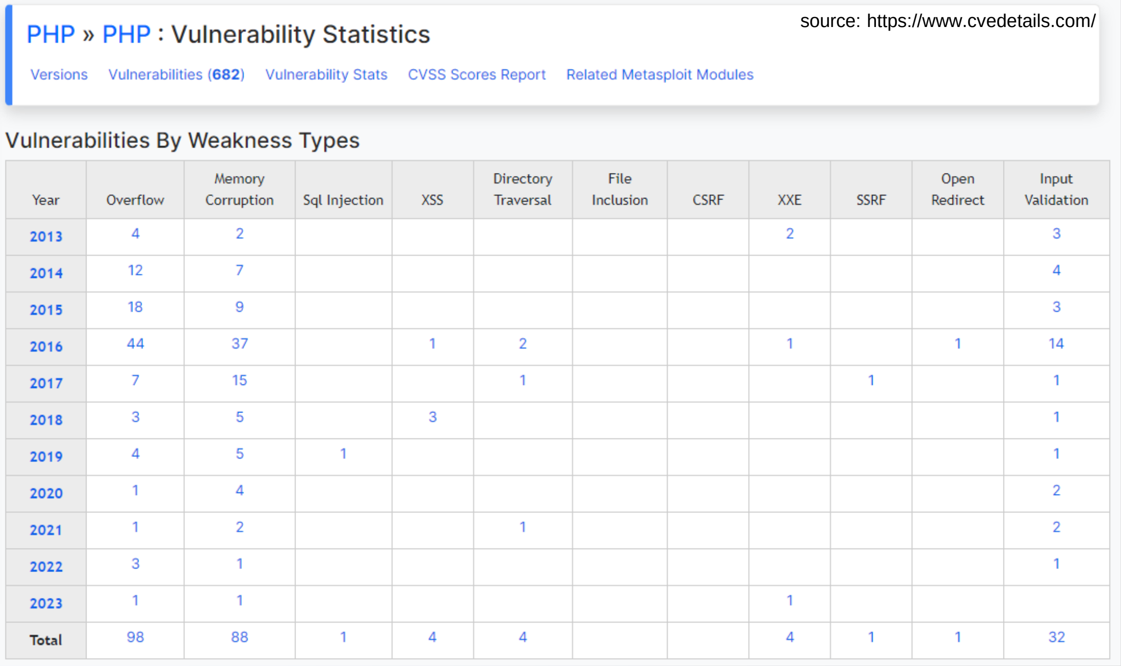


Microsoft » .net Framework : Vulnerability Statistics

Versions · Vulnerabilities (157) · Vulnerability Stats · CVSS Scores Report · Related Metasploit Modules

Vulnerabilities By Weakness Types

Year	Overflow	Memory Corruption	Sql Injection	XSS	Directory Traversal	File Inclusion	CSRF	XXE	SSRF	Open Redirect	Input Validation
2013	2										5
2014		1									3
2015	2	1		1				1			7
2016	1	1						1			1
2017											2
2018								1			2
2019	1				2	1					2
2020											2
2021											
2022											
2023											
Total	6	3		1	2	1		3			24



source: https://www.cvedetails.com/

PHP » PHP : Vulnerability Statistics

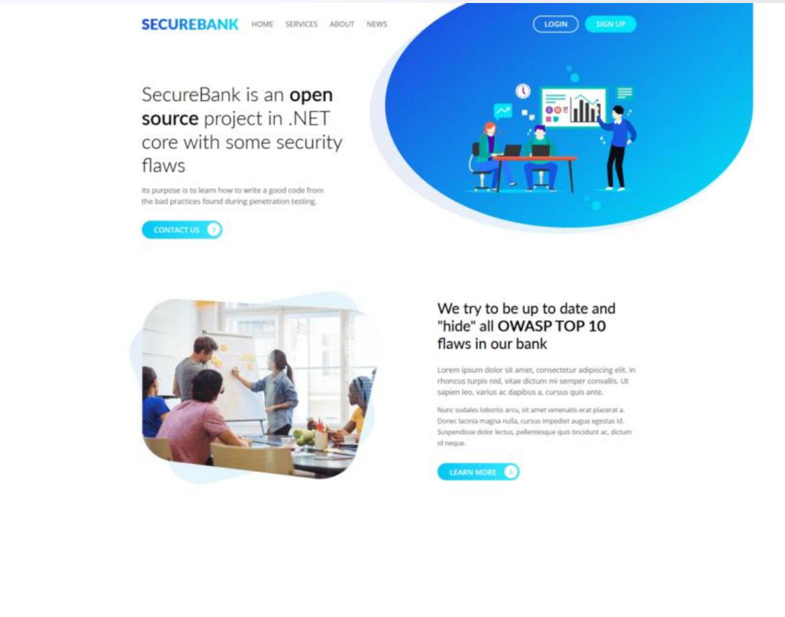
Versions · Vulnerabilities (682) · Vulnerability Stats · CVSS Scores Report · Related Metasploit Modules

Vulnerabilities By Weakness Types

Year	Overflow	Memory Corruption	Sql Injection	XSS	Directory Traversal	File Inclusion	CSRF	XXE	SSRF	Open Redirect	Input Validation
2013	4	2						2			3
2014	12	7									4
2015	18	9									3
2016	44	37		1	2			1		1	14
2017	7	15			1				1		1
2018	3	5		3							1
2019	4	5	1								1
2020	1	4									2
2021	1	2			1						2
2022	3	1									1
2023	1	1						1			
Total	98	88	1	4	4			4	1	1	32

(In)Secure bank

- OWASP project
  - https://github.com/ssrdio/SecureBank
- Microservice infrastructure
- Microservice architecture
- .NET Core
- OWASP Top 10
  - 2017
  - 2021
- Logical errors



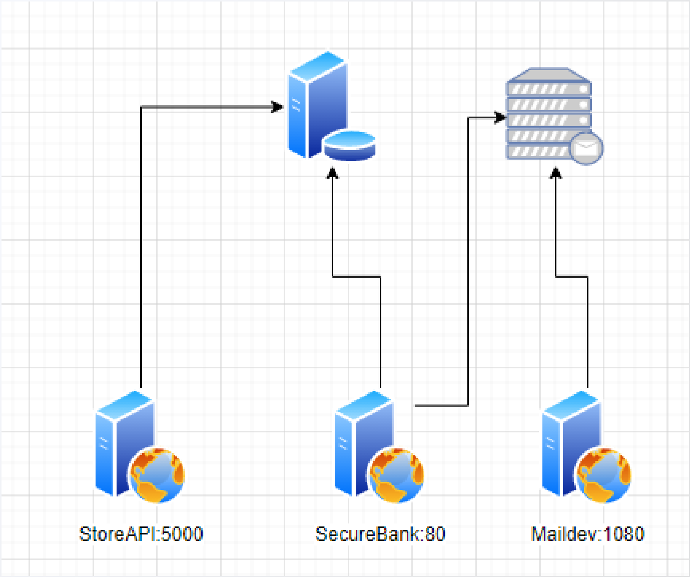
SecureBank is an **open source** project in .NET core with some security flaws

Its purpose is to learn how to write a good code from the bad practices found during penetration testing.

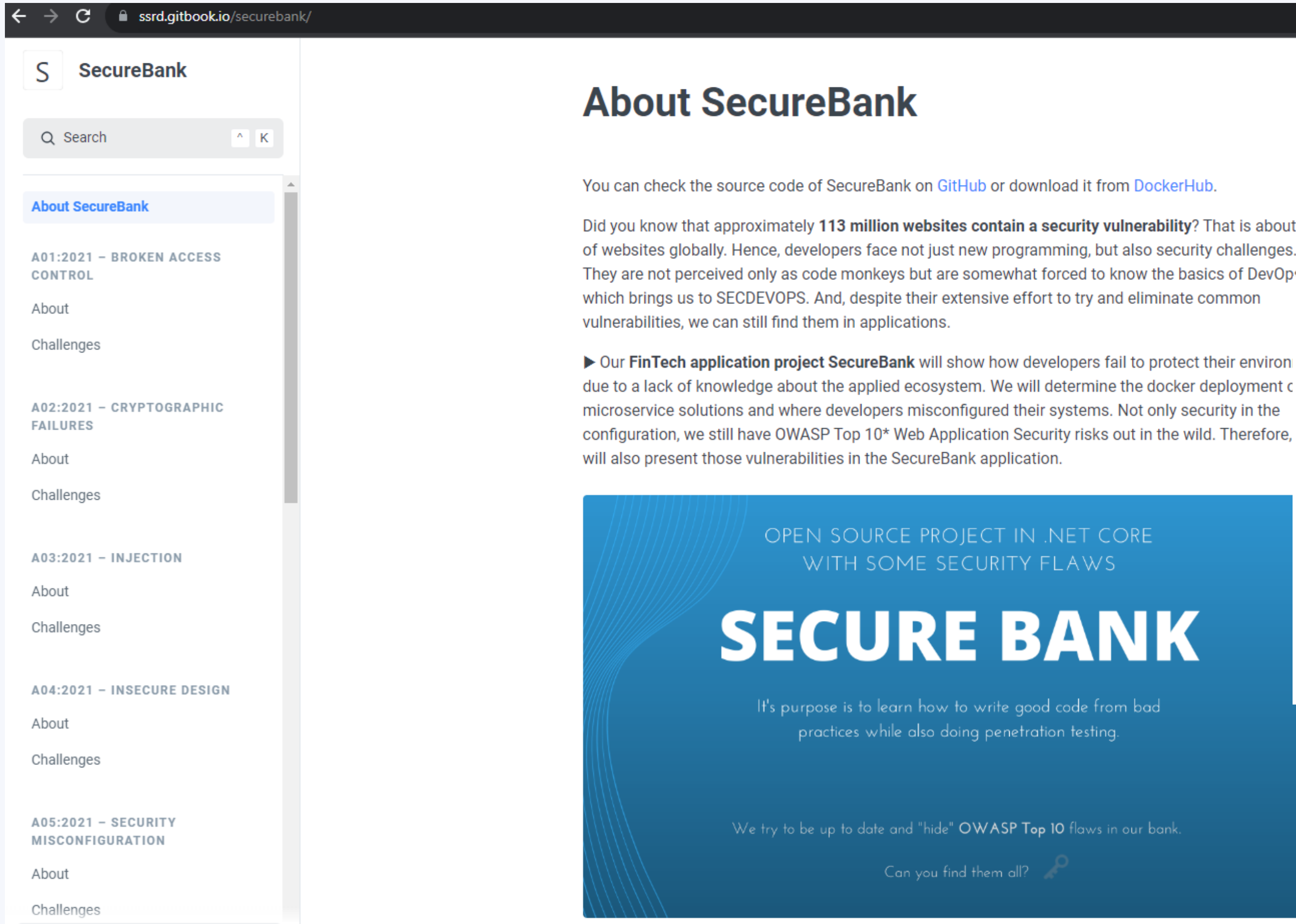
We try to be up to date and "hide" all **OWASP TOP 10** flaws in our bank

Lorem ipsum dolor sit amet, consectetur adipiscing elit. In rhoncus turpis nisl, vitae dictum mi semper convallis. Ut sapien leo, varius ac dapibus a, cursus quis ante.

Nunc sodales lobortis arcu, sit amet venenatis erat placerat a. Donec lacinia magna nulla, cursus imperdiet augue egestas id. Suspendisse dolor lectus, pellentesque quis tincidunt ac, dictum id neque.



How to



ssrd.gitbook.io/securebank/
SecureBank
Search
About SecureBank
A01:2021 – BROKEN ACCESS CONTROL
About
Challenges
A02:2021 – CRYPTOGRAPHIC FAILURES
About
Challenges
A03:2021 – INJECTION
About
Challenges
A04:2021 – INSECURE DESIGN
About
Challenges
A05:2021 – SECURITY MISCONFIGURATION
About
Challenges

About SecureBank

You can check the source code of SecureBank on GitHub or download it from DockerHub.

Did you know that approximately **113 million websites contain a security vulnerability**? That is about 6% of websites globally. Hence, developers face not just new programming, but also security challenges. They are not perceived only as code monkeys but are somewhat forced to know the basics of DevOps which brings us to SECDEVOPS. And, despite their extensive effort to try and eliminate common vulnerabilities, we can still find them in applications.

▶ Our **FinTech application project SecureBank** will show how developers fail to protect their environ due to a lack of knowledge about the applied ecosystem. We will determine the docker deployment c microservice solutions and where developers misconfigured their systems. Not only security in the configuration, we still have OWASP Top 10* Web Application Security risks out in the wild. Therefore, will also present those vulnerabilities in the SecureBank application.

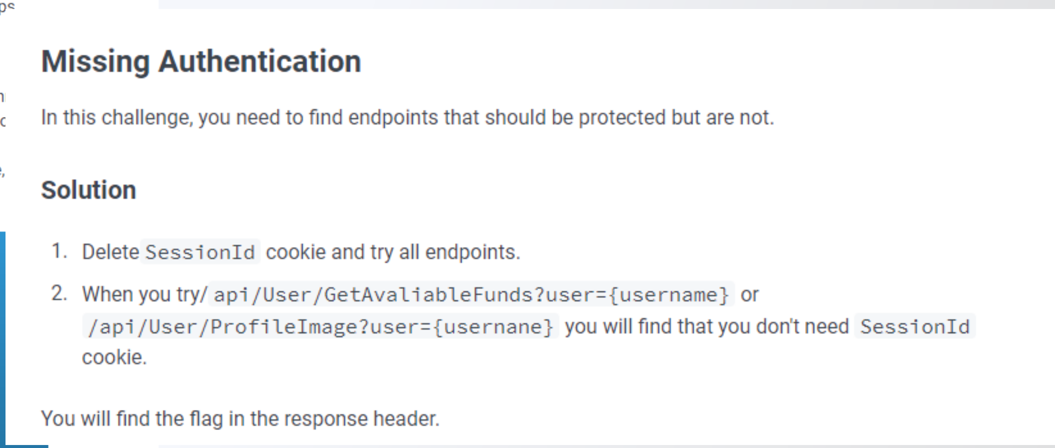
OPEN SOURCE PROJECT IN .NET CORE
WITH SOME SECURITY FLAWS

SECURE BANK

It's purpose is to learn how to write good code from bad practices while also doing penetration testing.

We try to be up to date and "hide" OWASP Top 10 flaws in our bank.

Can you find them all?



Missing Authentication

In this challenge, you need to find endpoints that should be protected but are not.

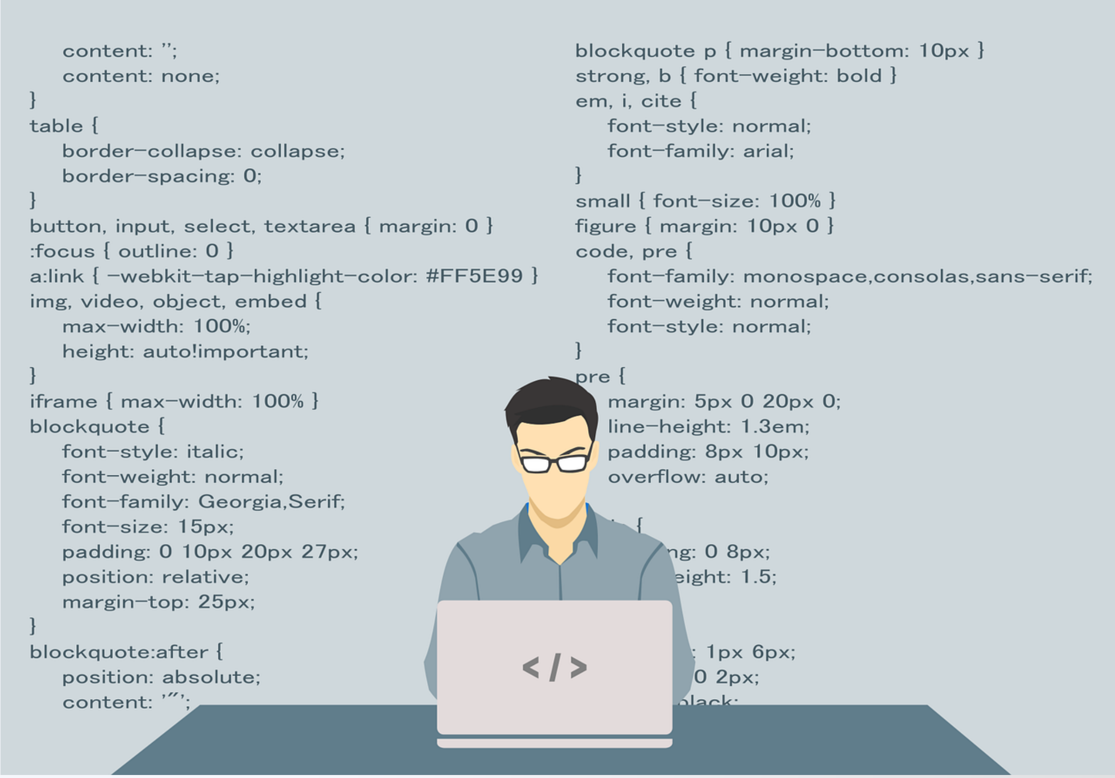
Solution

1. Delete `SessionId` cookie and try all endpoints.
2. When you try/ `api/User/GetAvaliableFunds?user={username}` or `/api/User/ProfileImage?user={usernane}` you will find that you don't need `SessionId` cookie.

You will find the flag in the response header.

Code analysis

- Static code analysis
  - Logic and technique
  - Execution path
  - Code review
- Dynamic code analysis
  - Complex issue
  - Microservice architecture
  - Live testing



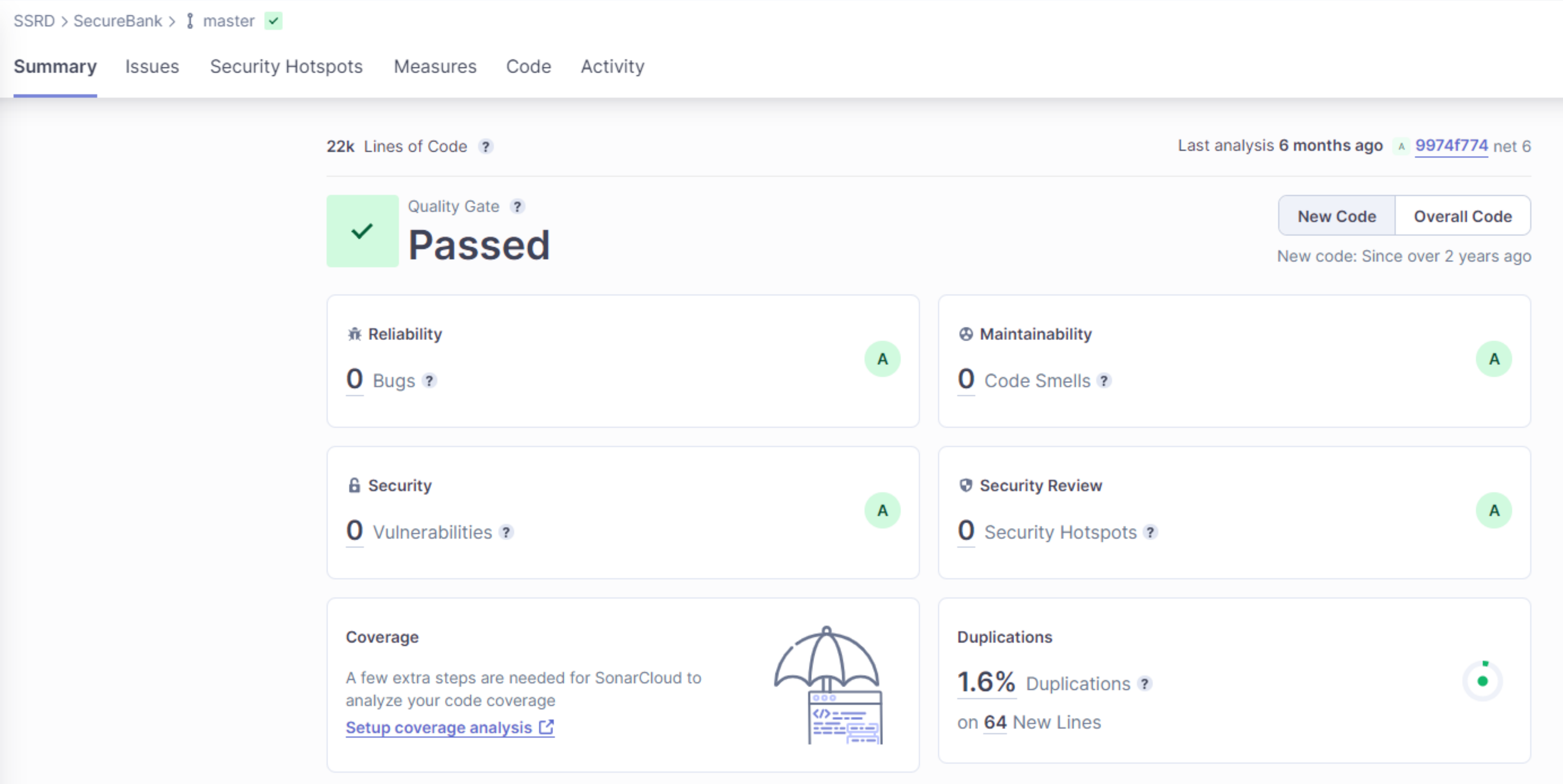
```
    content: '';
    content: none;
}
table {
    border-collapse: collapse;
    border-spacing: 0;
}
button, input, select, textarea { margin: 0 }
:focus { outline: 0 }
a:link { -webkit-tap-highlight-color: #FF5E99 }
img, video, object, embed {
    max-width: 100%;
    height: auto!important;
}
iframe { max-width: 100% }
blockquote {
    font-style: italic;
    font-weight: normal;
    font-family: Georgia,Serif;
    font-size: 15px;
    padding: 0 10px 20px 27px;
    position: relative;
    margin-top: 25px;
}
blockquote:after {
    position: absolute;
    content: '”';
```

```
blockquote p { margin-bottom: 10px }
strong, b { font-weight: bold }
em, i, cite {
    font-style: normal;
    font-family: arial;
}
small { font-size: 100% }
figure { margin: 10px 0 }
code, pre {
    font-family: monospace,consolas,sans-serif;
    font-weight: normal;
    font-style: normal;
}
pre {
    margin: 5px 0 20px 0;
    line-height: 1.3em;
    padding: 8px 10px;
    overflow: auto;

        {
      ng: 0 8px;
      eight: 1.5;

      : 1px 6px;
      0 2px;
      lack:
```

Secure bank



SSRD > SecureBank > master ✓

22k Lines of Code ?

Last analysis 6 months ago A 9974f774 net 6

Quality Gate ?

✓ Passed

New code: Since over 2 years ago

Reliability

0 Bugs ? A

Maintainability

0 Code Smells ? A

Security

0 Vulnerabilities ? A

Security Review

0 Security Hotspots ? A

Coverage

A few extra steps are needed for SonarCloud to analyze your code coverage

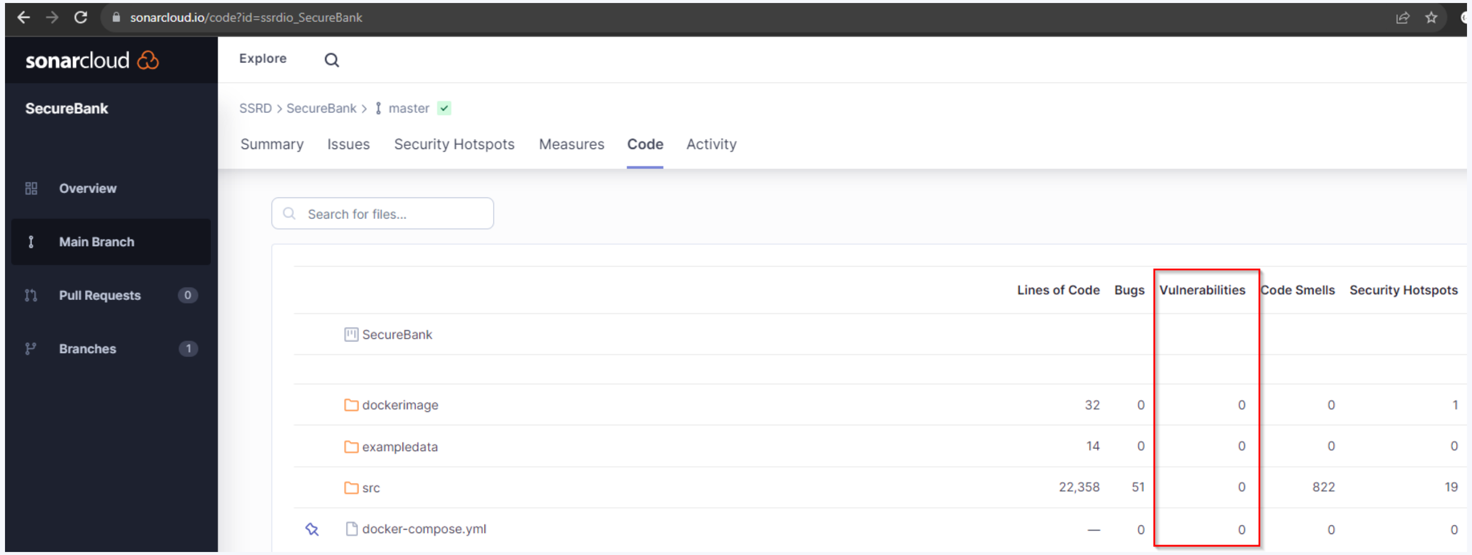
Setup coverage analysis

Duplications

1.6% Duplications ?

on 64 New Lines

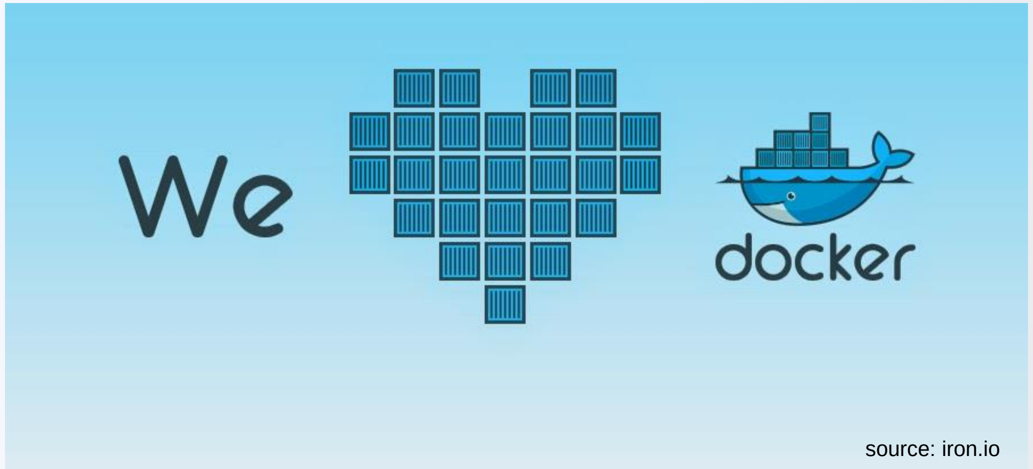
In theory secure by default



	Lines of Code	Bugs	Vulnerabilities	Code Smells	Security Hotspots
SecureBank					
dockerimage	32	0	0	0	1
exampledata	14	0	0	0	0
src	22,358	51	0	822	19
docker-compose.yml	—	0	0	0	0

Run & play

- Thx to Docker
  - It is possible!

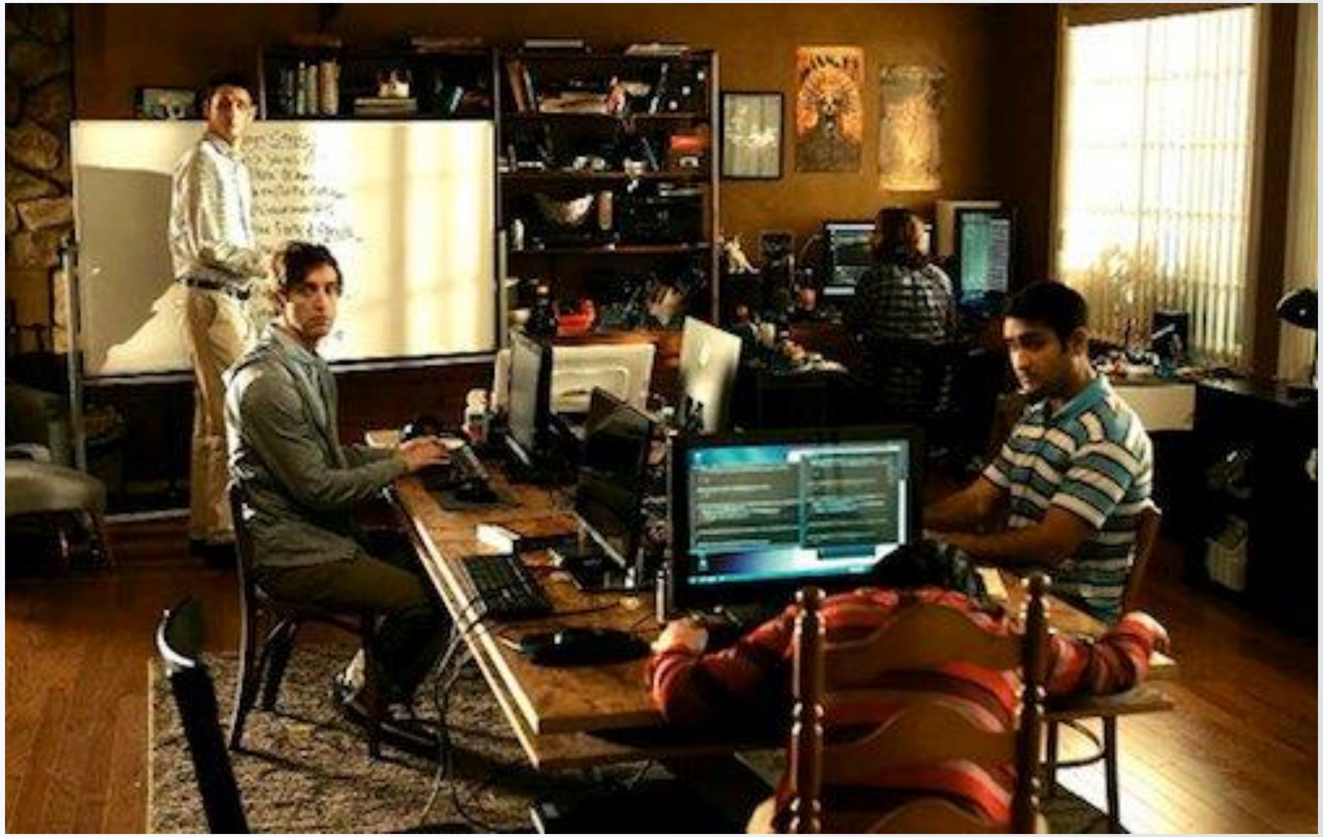


source: iron.io

```
docker run -d -p 80:80 -p 5000:5000 ssrd/securebank
```

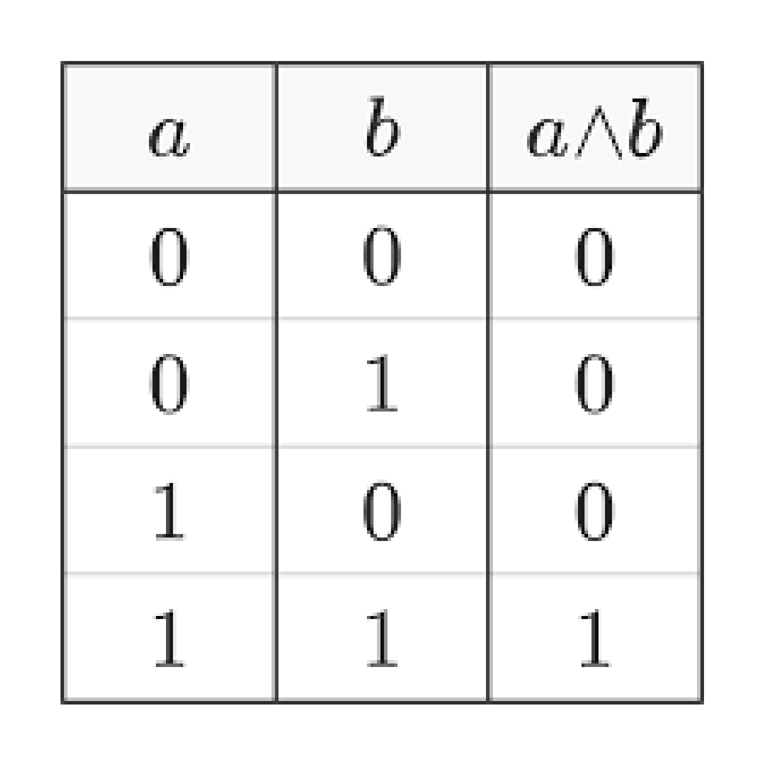
People

- CTO
- Frontend Engineer
- Backend Engineer
- Full stack Engineer
- DevOps Engineer
- Administrator
- Quality Engineer
- Security Engineer



People

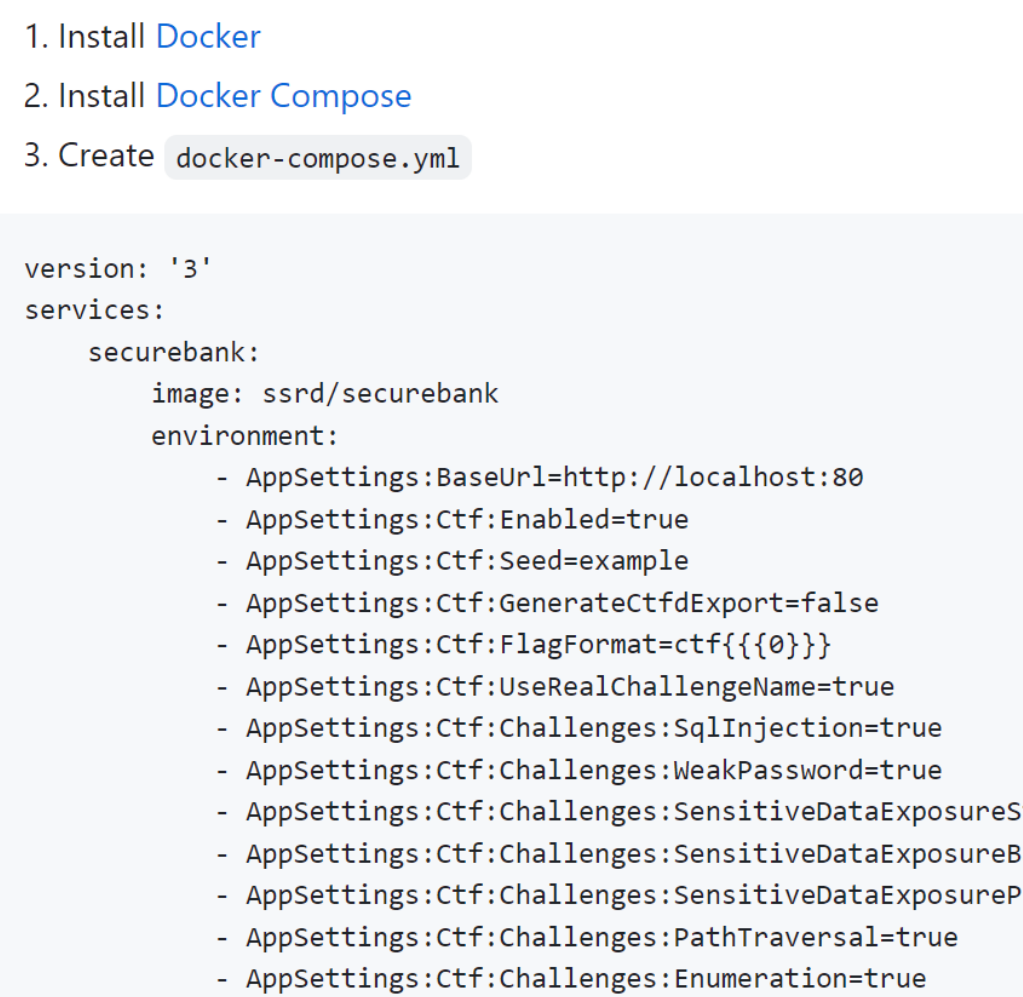
- CTO
- Frontend Engineer
- Backend Engineer
- Full stack Engineer
- DevOps Engineer
- Administrator
- Quality Engineer
- Security Engineer



a	b	$a \wedge b$
0	0	0
0	1	0
1	0	0
1	1	1

Train

- CTF

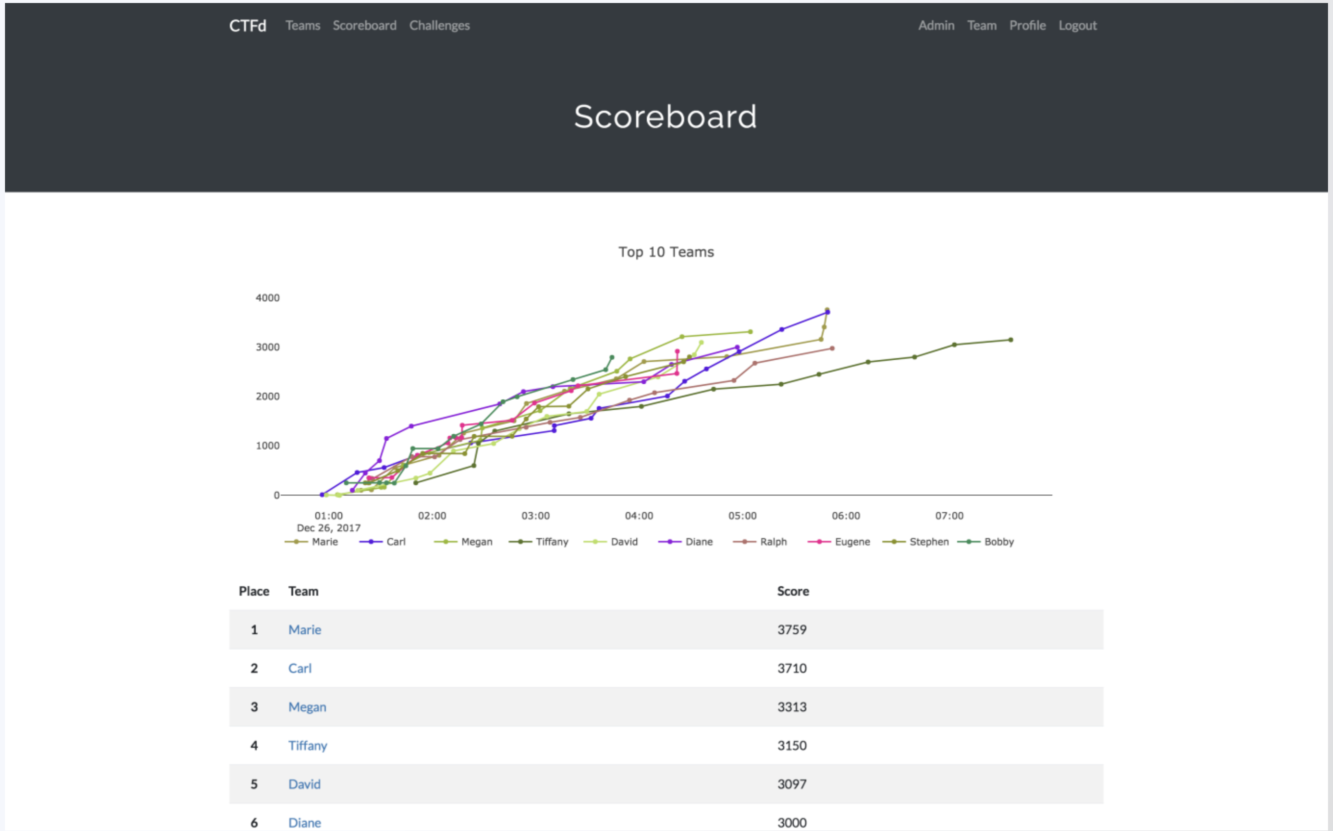


1. Install Docker
2. Install Docker Compose
3. Create `docker-compose.yml`

```
version: '3'
services:
    securebank:
        image: ssrd/securebank
        environment:
            - AppSettings:BaseUrl=http://localhost:80
            - AppSettings:Ctf:Enabled=true
            - AppSettings:Ctf:Seed=example
            - AppSettings:Ctf:GenerateCtfdExport=false
            - AppSettings:Ctf:FlagFormat=ctf{{{0}}}
            - AppSettings:Ctf:UseRealChallengeName=true
            - AppSettings:Ctf:Challenges:SqlInjection=true
            - AppSettings:Ctf:Challenges:WeakPassword=true
            - AppSettings:Ctf:Challenges:SensitiveDataExposureStore=true
            - AppSettings:Ctf:Challenges:SensitiveDataExposureBalance=true
            - AppSettings:Ctf:Challenges:SensitiveDataExposureProfileImage=true
            - AppSettings:Ctf:Challenges:PathTraversal=true
            - AppSettings:Ctf:Challenges:Enumeration=true
```

Competition

- CTFd



Scoreboard

Place	Team	Score
1	Marie	3759
2	Carl	3710
3	Megan	3313
4	Tiffany	3150
5	David	3097
6	Diane	3000

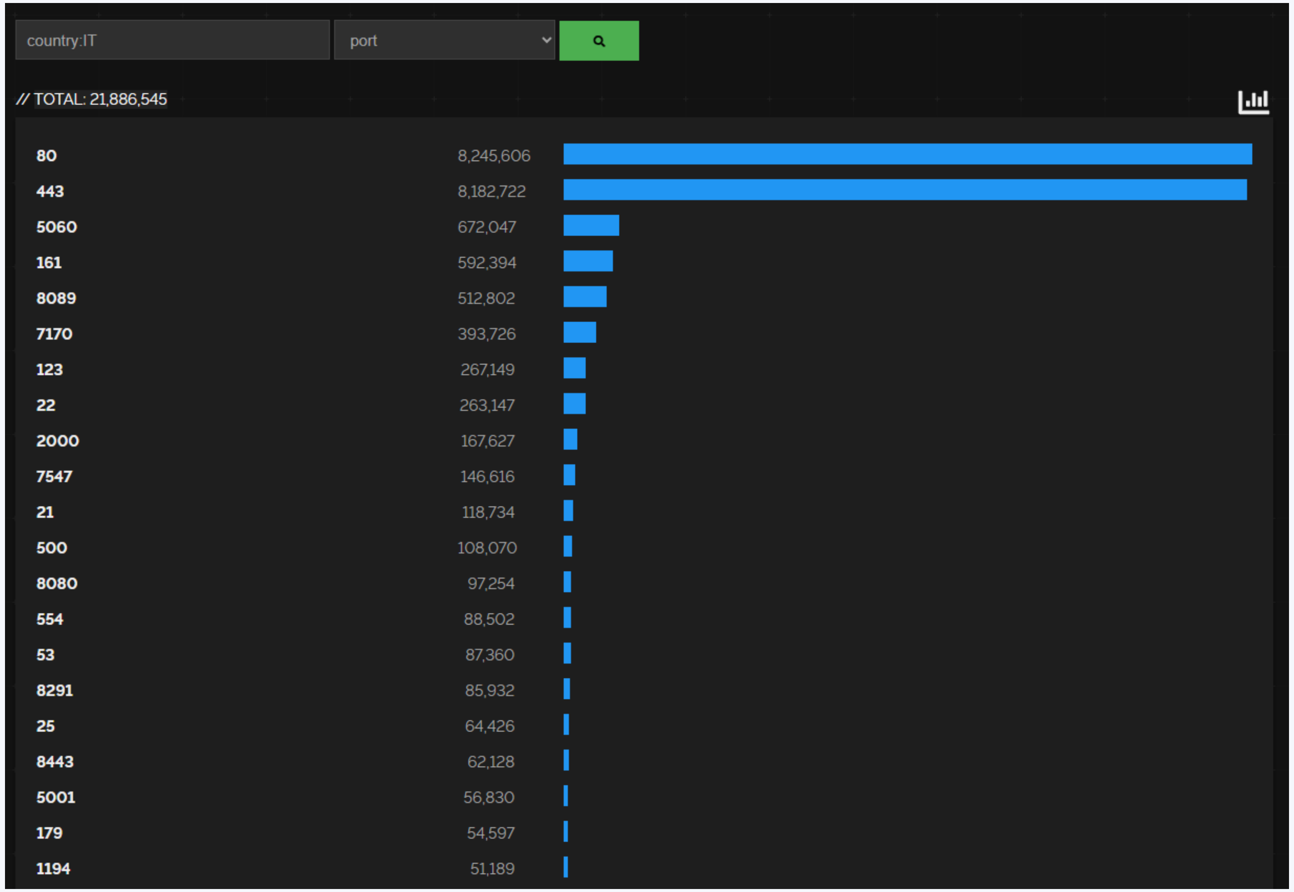
Moving forward

- CI/CD integration
- A/D
- Additional challenges
- Help appreciated

What we need to do?

- Evangelizing cyber security awareness
- Develop communities
- Write understandable security reports
- Help developers

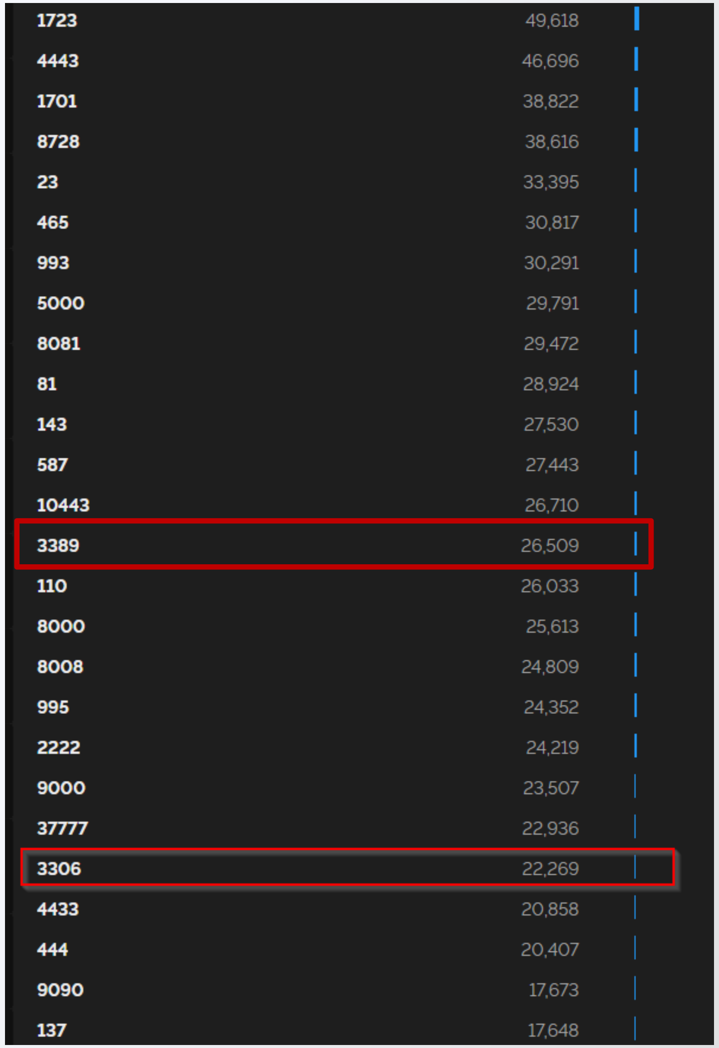
Italy in numbers



country:IT | port

// TOTAL: 21,886,545

80	8,245,606
443	8,182,722
5060	672,047
161	592,394
8089	512,802
7170	393,726
123	267,149
22	263,147
2000	167,627
7547	146,616
21	118,734
500	108,070
8080	97,254
554	88,502
53	87,360
8291	85,932
25	64,426
8443	62,128
5001	56,830
179	54,597
1194	51,189



1723	49,618
4443	46,696
1701	38,822
8728	38,616
23	33,395
465	30,817
993	30,291
5000	29,791
8081	29,472
81	28,924
143	27,530
587	27,443
10443	26,710
3389	26,509
110	26,033
8000	25,613
8008	24,809
995	24,352
2222	24,219
9000	23,507
37777	22,936
3306	22,269
4433	20,858
444	20,407
9090	17,673
137	17,648

Thank you to our sponsors

OWASP 2023

ITALY DAY