

Cyber Suite | HACKATHON26

Makropodis Ioannis
InQbit Innovations SRL



SASter

2nd CyberSuite hackathon
October 2026 | Paris, France



Co-funded by
the European Union

Co-Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or Digital Europe Programme. Neither the European Union nor Digital Europe Programme can be held responsible for them.



Agenda

01 Tool Information

05 Target Users

02 Purpose

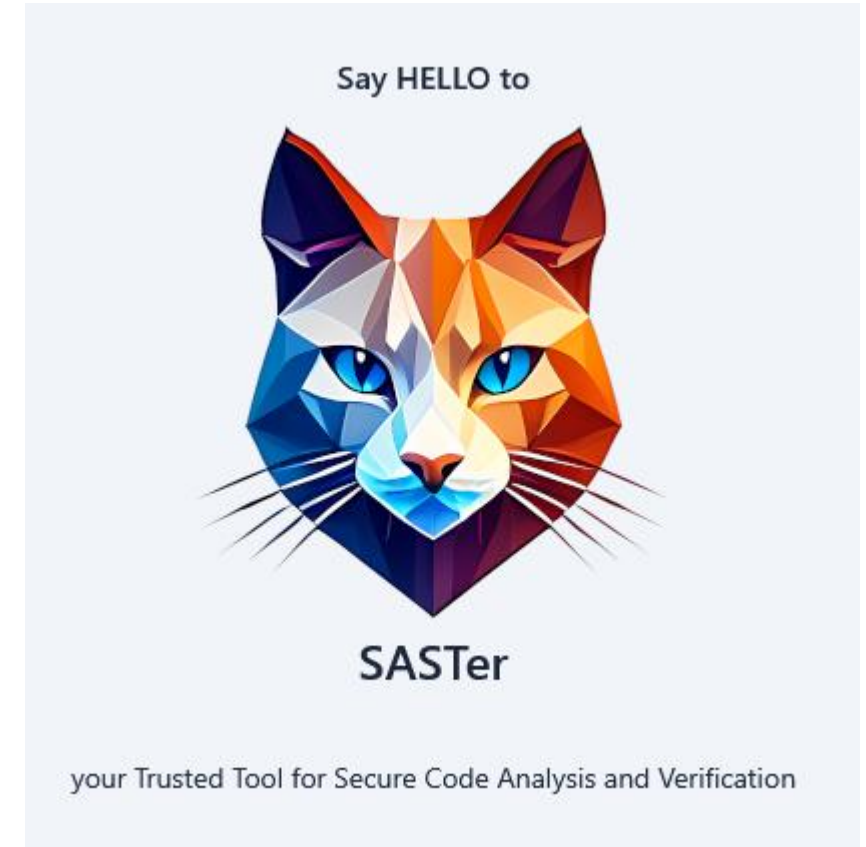
06 How is Tested?

03 Why SASTer?

04 Functionality



- SASTer is robust, easy-to-use Static Application Security Testing (SAST) tool designed for enhancing software security via static code analysis. It probes into the source code to identify potential security vulnerabilities, ensuring application safety. SASTer leverages a variety of open-source tools, utilizing their outputs for thorough security evaluations. It supports a wide range of technologies, including C/C++, Java, Python, and JavaScript, catering to various development settings.



- There is a growing challenge of software vulnerabilities, which cybercriminals exploit to launch attacks
 - Many SMEs struggle with cybersecurity due to a lack of resources, expertise, and access to enterprise-grade security solutions.
- Despite having limited resources, SMEs can take several steps to strengthen their cybersecurity posture. Adopting security tools like SASTer allows them to automate vulnerability detection, reducing the risk of security flaws in their software. SASTer helps mitigate these issues by detecting security flaws early, automating security checks, and providing actionable insights that allow developers to fix vulnerabilities quickly. By incorporating security analysis at the code level, it reduces the likelihood of software flaws being exploited once applications are deployed.

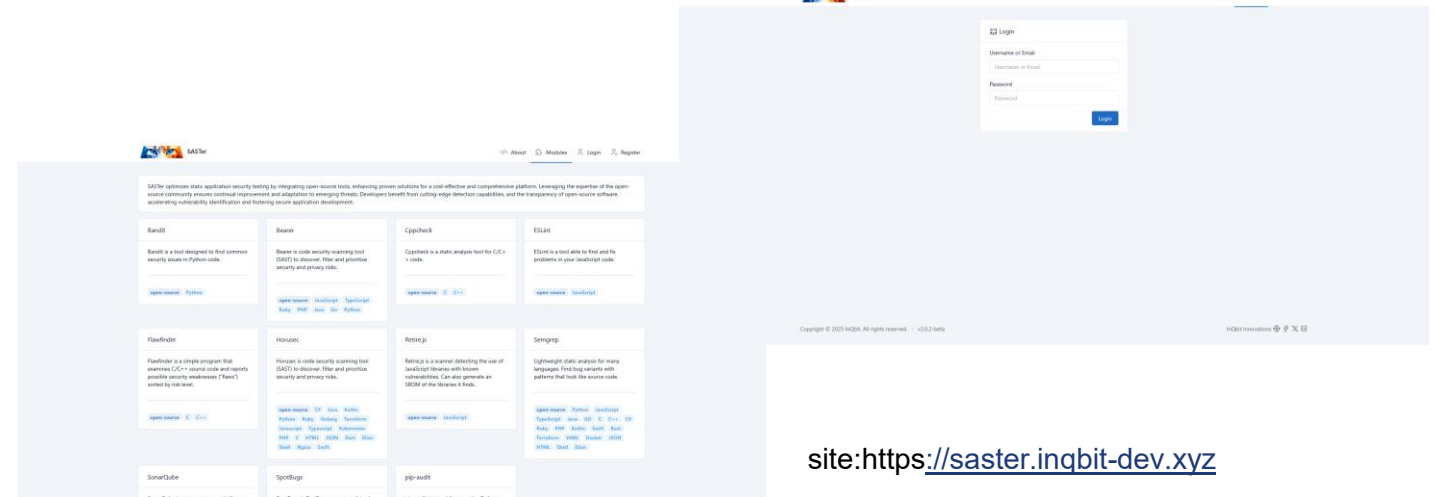


Why SASTer?

- Unifies **multiple open-source security tools** in one platform
- Builds on **proven solutions** for a cost-effective, comprehensive approach
- Harnesses the **open-source community** for continuous improvement
- Quickly adapts to **new and emerging threats**
- Offers developers **powerful detection capabilities** with full transparency
- Speeds up **vulnerability identification** and supports secure development



Tool	Languages	Purpose
Bandit	Python	Finds common security issues in Python code
Bearer	JS, TS, Ruby, PHP, Java, Go, Python	Security scanning for code & data risks
Cppcheck	C, C++	Static analysis for vulnerabilities
ESLint	JavaScript	Code linting and basic security checks
Flawfinder	C, C++	Identifies possible security weaknesses
Horusec	C#, Java, Kotlin, Go, etc.	Broad security scanning across languages
Retire.js	JavaScript	Detects vulnerable JS libraries
Semgrep	Multi-language	Pattern-based static analysis
SonarQube	Multi-language	Continuous code quality inspection
SpotBugs	Java	Finds bugs in Java code
pip-audit	Python (pip)	Scans dependencies for vulnerabilities

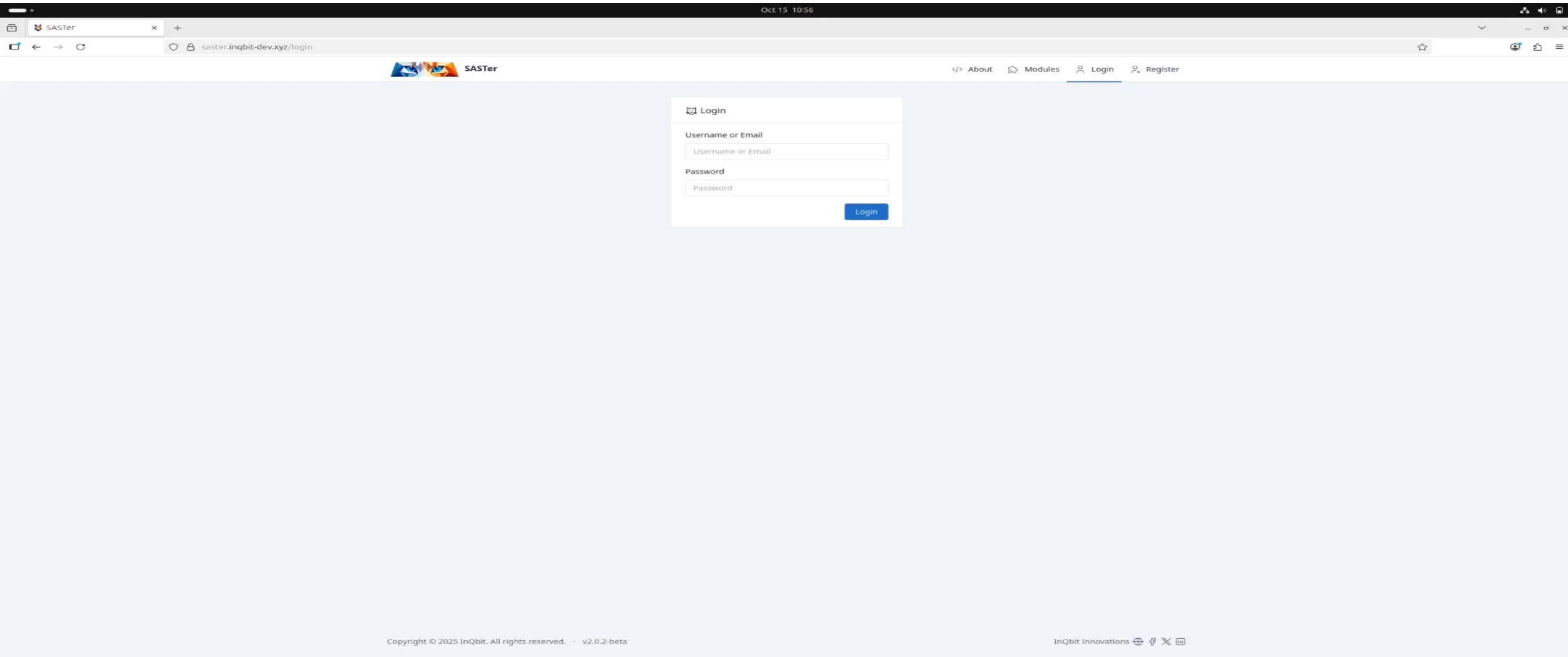


site:<https://saster.inqbit-dev.xyz>

- Saster also provides a RESTful HTTP API built with FastAPI that handles user authentication, session management, project lifecycle operations, and file storage. Each user may create named projects, upload individual files or ZIP archives and have those files safely extracted into a per-project workspace. The tool automatically computes folder statistics such as file counts, total size, and code-type tagging, persisting those metrics alongside project metadata in a SQLite database through SQLAlchemy.



How is tested?



Cyber Suite | HACKATHON26

Thank you for your attention!



InQbit Innovations SRL



Makropodis Ioannis



giannis.makropodis@inqbit.io



www.inqbit.io



Co-funded by
the European Union

Co-Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or Digital Europe Programme. Neither the European Union nor Digital Europe Programme can be held responsible for them.