FAST by Wallarm

Wallarm FAST

Automated Security Testing Integrated into CI/CD Pipeline

Improves security test coverage in minutes

- Tests single page applications and APIs
- Identifies security issues including OWASP Top10
- Tests for vulnerabilities and anomalies
- Gray-box tesintg
- Expandable with no coding

Aligns security and development into a unified pipeline

- CI tool agnostic; can be embedded anywhere

Automatically generates relevant focused tests

- Leverages existing tests that developers typically have in place
- Applies application -specific fuzzing
- Fine-grain control with policy

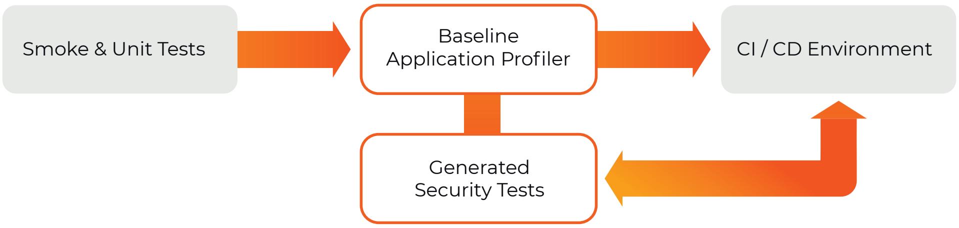
Wallarm Framework for Application Security Testing (FAST) enables ongoing security testing as a part of CI/CD. With continuous integration and continuous deployment applications are not shrink-wrapped software any more, applications are a service. To protect this service, security and test automation needs to become a continuous service as well.

Automatic Test Generation

- Tests for XSS, XXE, Path Traversal, SQLi, RCE and other OWASP Top 10 vulnerabilities
- Automated fuzzing of multiple parameters based on rules
- Gray Box testing - maintain session context
- Define and detect anomalies in addition to vulnerabilities
- Tests are generated from the existing functional tests / QA automation

Scalable Testing Environment

- Run security tests after every build
- Incorporate test flight results into release criteria
- Cloud-scale testing environment
- Up and running in minutes
- Collaborative test management
- Tests can be defined and started via API
- One line deployment in a Docker container



Wallarm FAST is focused on testing web application and API requests during the development and integration cycle. To implement this, FAST captures existing unit and smoke test traffic and uses their selection and structure as a basis to generate multiple security tests. This is accomplished by installing an easy-to-deploy Wallarm FAST node as a proxy to the test traffic.

For each baseline a set of tests is generated using fuzzing and the Wallarm threat database, which includes payloads for such common attacks as XSS, SQLi, RCE & Path Traversal (Ptrav) and XXE. Wallarm FAST then runs these sets of tests. The deployment of the proxy and the actual TestRun are typically automated via CI/CD environment and triggered by specific events, such as build completion. Wallarm FAST is designed to be a flexible test environment and provide Test Automation As A Service (TAaaS).

It is stack independent and will test applications developed in .NET, Java, Python, Ruby, PHP, and other development environments. Tests can be run locally or from the Wallarm Cloud service allowing DevOps full configuration control without having to worry about deployment environment, scale, or flexibility.

We support the following protocols (including nested protocol) for Deep Packet Inspection:

HTTP/2.0, REST, JSON, COMET, XML, SOAP, Base64, GZIP, VIEWSTATE, PHP (unserialize).

The framework is expandable without coding. Via a specialized YML-based definition language, (DSL), customized application-specific detects can be added to the scope of testing.

FAST
by Wallarm

Automate security within CI/CD

Wallarm FAST automatically generates a suite of tests for your application using our unique fuzzing technology and re-captured hacker intelligence.

Test Automation as a Service (TAaaS)

Incorporate security testing into your CI/CD process and rely on testing scalability and availability provided by Wallarm FAST with almost zero integration effort.

DevOps focus on business logic

Help the security team execute control of security while the application is still in development without slowing down the development process. The security team defines the policy the DevOps run automated test

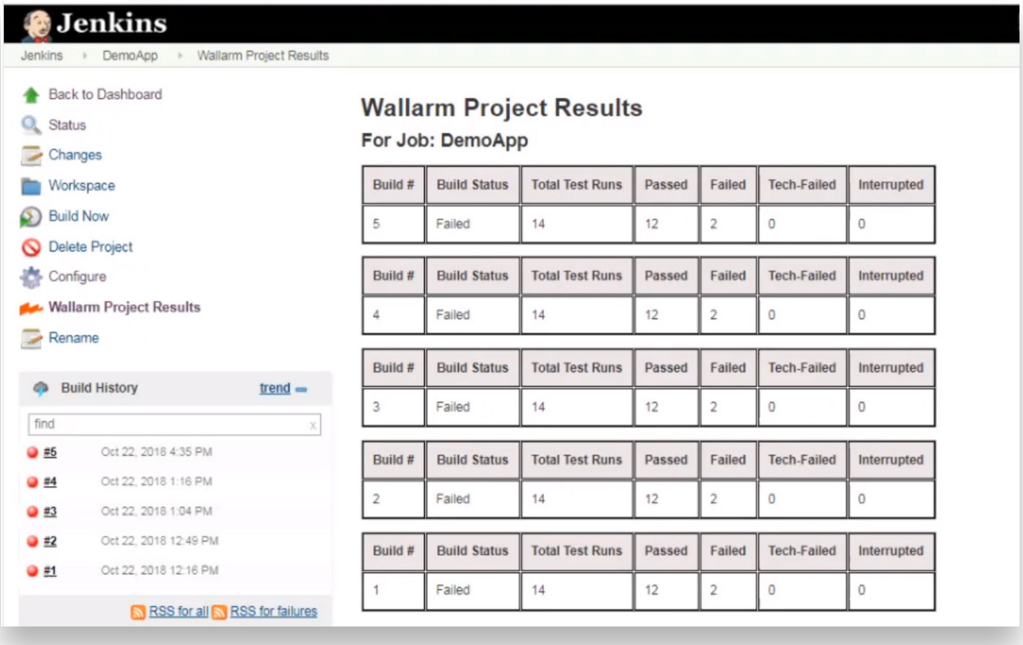
Sign up for a free trial

fast.wallarm.com/signup

wallarm.com
(415) 940–7077

Automation that doesn't slow down CI/CD

FAST automates security testing of the APIs and single-page apps in a focused time effective way by taking existing functional tests and applying security payloads.



#5 Oct 22, 2018 4:35 PM
#4 Oct 22, 2018 1:16 PM
#3 Oct 22, 2018 1:04 PM
#2 Oct 22, 2018 12:49 PM
#1 Oct 22, 2018 12:16 PM

Wallarm Project Results

For Job: DemoApp

Build #	Build Status	Total Test Runs	Passed	Failed	Tech-Failed	Interrupted
5	Failed	14	12	2	0	0

Build #	Build Status	Total Test Runs	Passed	Failed	Tech-Failed	Interrupted
4	Failed	14	12	2	0	0

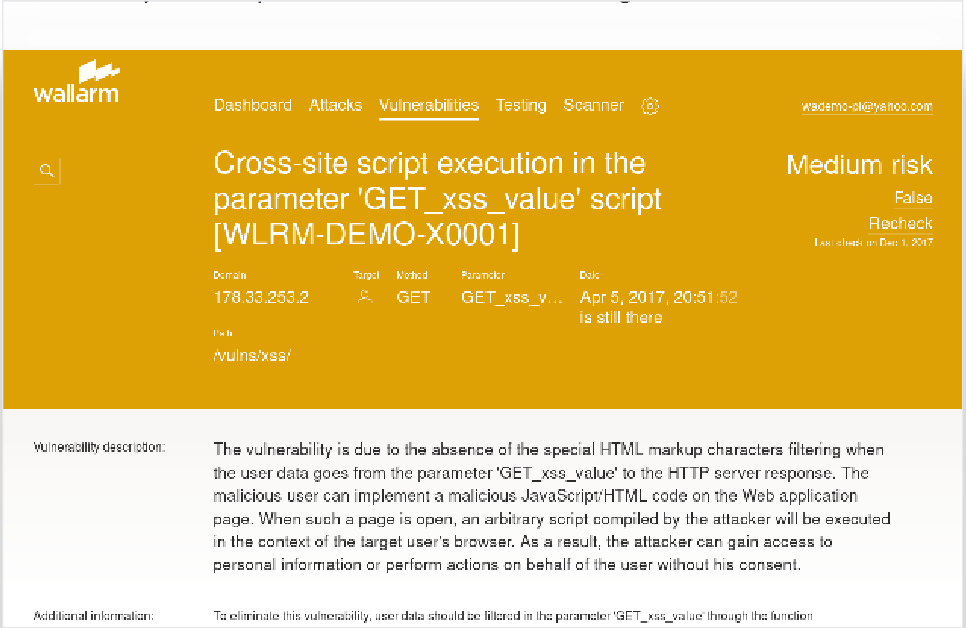
Build #	Build Status	Total Test Runs	Passed	Failed	Tech-Failed	Interrupted
3	Failed	14	12	2	0	0

Build #	Build Status	Total Test Runs	Passed	Failed	Tech-Failed	Interrupted
2	Failed	14	12	2	0	0

Build #	Build Status	Total Test Runs	Passed	Failed	Tech-Failed	Interrupted
1	Failed	14	12	2	0	0

Actionable Results

Wallarm FAST makes test results actionable. Search all the test cases and test run results by time, tag or TestRunID to drill into more details. Quickly see which APIs may present a problem and examine a sample exploit for every vulnerability or incorporate the information into regression tests.



Cross-site script execution in the parameter 'GET_xss_value' script [WLRM-DEMO-X0001]

Last check on Dec 1, 2017

Domain: 178.33.253.2
Target
Method: GET
Parameter: GET_xss_v…
Date: Apr 5, 2017, 20:51:52 is still there
Path: /vulns/xss/

Vulnerability description: The vulnerability is due to the absence of the special HTML markup characters filtering when the user data goes from the parameter 'GET_xss_value' to the HTTP server response. The malicious user can implement a malicious JavaScript/HTML code on the Web application page. When such a page is open, an arbitrary script compiled by the attacker will be executed in the context of the target user's browser. As a result, the attacker can gain access to personal information or perform actions on behalf of the user without his consent.

Additional information: To eliminate this vulnerability, user data should be filtered in the parameter 'GET_xss_value' through the function