PEACH API SECURITY
AUTOMATING API SECURITY TESTING
Table of Contents

Introduction.................................................................................................................................................................................. 3
Industry Trends.................................................................................................................................................................................. 3
API growth..................................................................................................................................................................................... 3
Agile and Continuous Development Frameworks..................................................................................................................... 4
Gaps in Tooling.................................................................................................................................................................................. 4
Peach API Security............................................................................................................................................................................ 4
What we do..................................................................................................................................................................................... 4
How we do it..................................................................................................................................................................................... 4
Detailed Fault Data.......................................................................................................................................................................... 5
Conclusion.................................................................................................................................................................................... 6
About Peach Tech.............................................................................................................................................................................. 6
**Introduction**

Modern agile development frameworks have changed the way engineering teams produce products. These frameworks, coupled with the widespread adoption of APIs, have led to new requirements for security testing tools.

As a leader in creating automated security testing solutions, Peach Tech recognizes the importance of automating security testing for this new industry. Our products deliver security testing to organizations that is quicker, more scalable, and less expensive than traditional testing.

This paper outlines the approach Peach API Security uses to test modern APIs. Used by organizations worldwide, from large enterprises to small developer teams, Peach API Security automates security testing of APIs by turning DevOps teams into DevSecOps teams.

"As more enterprises roll out new business models and digital initiatives, we see a growing number of APIs and API traffic volume that needs to be secured, managed, scaled, and analyzed. We predict a further ten-fold increase."

Apigee -2016 State of APIs Report

**Industry Trends**

Many modern enterprises rely on REST and SOAP based APIs to create, maintain, and transport their critical business and customer data. Modern web and mobile applications are often simple UIs that enable users to interact with a backend powered by APIs. Some companies’ entire business models are to design and layer APIs to perform useful services. As the adoption of APIs continues to grow, so do the risks for organizations who don’t actively test the security of their solutions.

**API growth**

The growth of web technology and IoT companies has led to an explosion in the number of APIs being used. At the time of this writing, there are over 17,000 public APIs available, with many more being added daily. Continued growth and adoption of APIs is expected over the next decade.

**APIs**

- Mobile Applications
- Web Applications
- Desktop Applications
- Browser Applications
- Microservices
- IoT
- Embedded Devices

**Agile and Continuous Development Frameworks**

The industry has broadly transitioned to Agile and continuous development frameworks. Under these frameworks, products receive small, frequent updates daily or weekly rather than major quarterly or annual product releases. These frameworks rely on continuous integration (CI) systems to facilitate each build. While the product’s efficacy and flexibility are improved by these small releases, each new build introduces the potential for security vulnerabilities.

**Gaps in Tooling**

The security tools currently used to test APIs are ineffective, costly, and slow to deploy. Until recently, there has not been a cost-effective tool capable of testing APIs for security issues throughout the entire development lifecycle. Rather, organizations have been forced to rely on manual tools and experienced penetration testers. These solutions have been costly, slow, and occur late in the development process.

While analyzing the current landscape of security testing tools in the space, several gaps emerged:

**API Specific Testing**

Many existing solutions, such as BURP, are generalist security tools -- not designed to perform security testing on modern APIs. These web-scanner style tools work by pointing to a single service, API, or endpoint. The tool then crawls that service and captures the traffic that is sent between the API and client. By analyzing this data, the tool returns a pass or fail result. Each vulnerability then needs to be manually located and verified by the penetration tester before it can be sent to a developer.

These solutions lack support for many of the complexities of modern APIs. The older point and shoot tools require users to manually configure the tool for each API endpoint before they can be tested. This laborious configuration step wastes time and leads to mistakes in testing. Additionally, these tools do not support modern authentication schemes used by most APIs and applications. This gap in support leads to poor code coverage, putting organizations at risk.
Lack Developer Integration
Security tools should work with the tools developers already use to create and manage their products. Current security tools lack basic integrations, such as the ability to capture and store log files, integrate with CI build systems, or send fault findings to common bug tracking software solutions.

These tools, which were intended to be used by security experts and pen testers, require manual efforts to test for, verify, and log vulnerabilities. It is often cost prohibitive for developer teams to invest the time and resources to learn these tools.

End of Development
Because of the difficulty developer teams face in integrating existing security tools into their workflows, testing of products is often performed by security professionals late in the development process. Bugs are much more expensive to fix the later they are found, as they are often diffused throughout the product by this point of development.

The time and cost to manually test products frequently encourages testing to occur only once per major development cycle rather than for every new build of the product. This results in intermediate builds of products being put into production without proper testing.

Preventing Releases
Because existing security tools were not designed to work with the CI systems used by modern engineering teams, they do not have the ability to stop builds with known vulnerabilities from being released to users. Additionally, the time it takes to find and fix all vulnerabilities when they are discovered late can cause ship date delays.

What We Do
Peach API Security, a dynamic application security testing tool, was designed to fill the gaps left by other API security testing tools. The tool is designed to automate testing of APIs without interrupting an engineering team’s workflow.

Peach API Security was designed with three key tenants in mind:

API Specific Security Testing
Our tool tests against the OWASP Top-10 vulnerability list. This list was created to highlight vulnerabilities found in modern applications.

It is specifically intended to handle the complexity of modern REST and SOAP based APIs and services. This includes the

abilities to handle both modern authentication schemes, as well as complex interactions between multiple APIs and services.

**OWASP Top-10**

- **Injection**
- **Broken Auth & Session Management**
- **Cross-Site Scripting (XSS)**
- **Insecure Direct Object Reference**
- **Security Misconfiguration**
- **Sensitive Data Exposure**
- **Missing Function Level Access Control**
- **Cross-Site Request Forgery (CSRF)**
- **Using Known Vulnerable Components**
- **Unvalidated Redirects and Forwards**

Automated Testing
Peach API Security automates security testing of APIs by leveraging the work engineering teams are already doing. Integrations with automation testing frameworks enable Peach to convert automation or unit tests into security tests. Leveraging automation tests moves security testing earlier in the development lifecycle where fixing issues is cheaper and easier.

Don’t Break Developer Workflow
Acting as a build step in the most popular CI systems, Peach API Security runs a series of security checks on the target service. When faults are discovered, vulnerable builds are automatically prevented from being put into production.

All interactions occur from within the tooling developers are already familiar with. Organizations don’t have to invest time and resources altering team’s workflows or teaching their engineers to use new security tools. Security is baked into every build automatically.

How we do it
Peach API Security helps organizations automate security testing of their applications by turning DevOps teams into DevSecOps teams.

Checks
Every run of Peach API Security performs a series of checks against the OWASP Top-10 vulnerabilities list. This list includes some of the most pervasive and harmful security flaws in modern applications.
Fuzzing

Peach API Security also uses a modified version of the Peach Fuzzer testing engine to mutate fields sent to the backend API. Mutating valid messages can uncover additional security vulnerabilities that are not covered by the OWASP Top-10.

Test Case Generation

Engineering teams commonly use automation or unit tests to ensure that their products behave as expected. These automation tests are designed to provide good code coverage and pass through the product’s authentication schemes.

Peach API Security acts as a man-in-the-middle proxy between a traffic generator that sends valid automation tests and the target service.

By converting valid automation tests into mutated security tests, many security vulnerabilities can be uncovered. Repeatedly sending mutated tests to the target service replaces the manual work previously required of other tools and allows for more robust security coverage.

CI Integration

Integrations with the most common CI pipeline build systems simplifies testing for engineering teams. Peach acts as a step in the build pipeline, automatically launching each time a new build is kicked off.

When a vulnerability is detected, the build is automatically flagged and will not be deployed. This prevents a vulnerable build from being released to production.

Developer teams using agile or continuous development methodologies are already familiar with CI systems. Peach’s tight integrations enable all interactions, including review of fault findings, to occur directly from within the CI tool.

Testing Profiles

Recognizing that speed and shipping deadlines are critical for teams that push builds multiple times a day, Peach includes several configurable testing profiles. This allows users to balance security testing coverage with ship deadline requirements. Each testing profile is configurable, allowing teams to include or exclude certain checks or modes of testing.

<table>
<thead>
<tr>
<th>Purpose</th>
<th>Quick Turn-arounds</th>
<th>NIGHTLY Overnight Testing</th>
<th>WEEKLY Thorough Testing</th>
<th>FULL Full, Single-shot testing</th>
</tr>
</thead>
<tbody>
<tr>
<td>Example</td>
<td>Multiple Builds per Day</td>
<td>Single Build per Day</td>
<td>Deeper Testing on Weekend</td>
<td>Major Product Releases</td>
</tr>
<tr>
<td>Checks</td>
<td>Limited</td>
<td>Full</td>
<td>Full</td>
<td>Full</td>
</tr>
<tr>
<td>Fuzzing</td>
<td>None</td>
<td>Limited</td>
<td>Full</td>
<td>Full</td>
</tr>
</tbody>
</table>

Detailed Fault Data

Instant and actionable fault data enables engineers to spend less time finding and fixing bugs and more time developing robust products for customers. Peach API Security monitors HTTP status codes, target logs, and response bodies to detect faults. In addition to pre-configured monitoring schemes, there is support for customization of monitoring to look for specific response messages which should be handled as a fault.

Fault Results

Findings include the information required for a developer to fix an issue quickly.

All fault findings are viewable from within the CI system the engineering team uses. Integrations with bug tracking software such as Jenkins or Bugzilla allow teams to manage and track issues.
Each fault result includes the following information:

- Operation/Parameter that Caused Fault
- Check/Assert that Failed
- Description of Failed Check/Assert
- Link to CVE/OWASP Information
- Exploitability/Impact Level
- Copies of Request and Response

**False Positives**

One common shortfall of existing automated security testing solutions is management of false positives. It takes considerable time to manually sort through which faults are valid bugs and which can be safely ignored.

Peach API security allows users to customize how faults are managed, so that false positives do not slow development. When a failure is determined to be a false positive or a team decides not to fix an issue, it can be added to a list of ignored failures.

Future builds will still test for the fault, but will not report it in ticket management systems or block builds from deploying.

**Conclusion**

Peach API Security gives organizations an automated, scalable tool for security testing of APIs. The product is purposely built to integrate into the workflow and tooling of engineering teams that use agile development frameworks. Peach API Security turns DevOps teams into DevSecOps teams.

**About Peach Tech**

Peach Fuzzer, LLC is a leader in developing Fuzz Testing based security tools. By providing automated, scalable, easy-to-use security testing platforms, we help the world's leading technology companies secure their products by discovering unknown vulnerabilities.