NETSPI™

Do You Have Your Own API Penetration Test Checklist?

5 Tips to Get You Started

API Penetration Testing is a specific type of application penetration test for comprehensive evaluation of the security of APIs. The use of APIs is increasing, and they tend to have larger attack surfaces, making them an increasingly popular pathway for threat actors to target. NetSPI's API Penetration Testing checklist prepares your team with a quick-hitting guide to prioritize API security.

✓ 1. Create Testing Categories

Grouping tests into logical categories can make it easier to build and maintain checklists over time. Many people look to the OWASP Top 10 for guidance, and while that is a good resource, NetSPI recommends you choose categories that will help meet your specific goals.

Common categories that NetSPI recommends:

- Attacking API authentication mechanism(s)
- Identifying access control weaknesses
- API server security configuration testing
- Analyzing exposed information to identify excessive data exposure
- API endpoint/method/parameter fuzzing
- Identifying server-side request forgery (SSRF) issues
- Rate limiting functionality testing

✓ 2. Create Baseline Tests

There are fewer resources published related to API testing and fewer still related to API pentest checklists. While no checklist will cover every scenario for all APIs, creating tests that cover core technologies and processes used during most API penetration tests will save you time and help ensure coverage of potential vulnerabilities.

Online resources to get you started:

- OWASP API Security Project
- OWASP Rest Assessment Cheat Sheet

Project Specific Information

A checklist serves as a place to store required procedures for engagements. It can also be a valuable place to store information discovered before, during, and after a test. A dynamic checklist creates a single source of information related to the test.

This information can also include:

- Connection information
- Application overview notes
- Application contacts
- Application-specific testing requirements
- Why certain procedures were skipped/modified
- Relevant notes for future testers

✓ 3. Link to External Content

Trying to fit every verbose testing procedure into your checklist may not be plausible but including links to relevant information can help you. NetSPI recommends limiting procedures to the most common scenarios and linking to the outliers. Including a small summary with resources and any troubleshooting that occurred during the first use can greatly increase the speed with which they can be used.

✓ 4. Choosing a Testing Checklist Platform

Choosing the right checklist solution can help free up more time for digging into vulnerabilities and unique attack scenarios specific to the in-scope APIs. There are several checklist platforms available on the internet that can be leveraged to create and maintain a list of common tasks. NetSPI recommends using one that has the capability to tie the tests to the findings that will ultimately turn into the report or ticket for the business owner. This will help reduce redundant tasks and increase the speed at which your team can get results to the people who need them.

✓ 5. Track and Remediate

The whole point of testing for API vulnerabilities is to fix them before someone else can take advantage of them. Having an established process for delivering identified vulnerabilities to the right people should be your first step at the end of your test.

Bug Bounty Hunters

Make sure to follow the affected vendor's acceptance channel – or work through a broker like HackerOne and BugCrowd.

Internal Penetration Testers

Make sure to work with internal business units to ensure the vulnerability is reported and remediated in accordance with the company's policy. NetSPI recommends having a checklist and a reporting process that integrates into ticketing systems – like NetSPI Resolve™.

To learn more about NetSPI's comprehensive offensive security solutions, visit www.netspi.com