Analisis Efektivitas Fuzzing Tools dan Pustaka Validasi Input pada Web API = Analysis of Effectiveness of Fuzzing Tools and Input Validation Libraries on Web APIs

Danar Gumilang Putera, author

Abstrak

Teknologi Web API sudah banyak diterapkan di berbagai infrastruktur aplikasi karena teknologi ini memungkinkan berbagai layanan aplikasi yang berbeda bisa saling berinteraksi dan berkomunikasi melalui media jaringan. Web API memungkinkan aplikasi untuk saling berbagi fungsionalitas dan data dengan aplikasi lain, menjadikannya teknologi yang paling banyak digunakan untuk integrasi antar infrastruktur. Terlepas dari manfaat Web API, hal ini bukannya tanpa masalah keamanan. Banyak kerentanan yang muncul akibat kesalahan konfigurasi atau mekanisme keamanan yang tidak memadai, yang dapat dicegah dengan melakukan pengujian fungsionalitas. Salah satu pengujian fungsionalitas yang penting untuk dilakukan adalah fuzzing. Fuzzing adalah metode pengujian untuk mengidentifikasi kerentanan yang muncul dari kesalahan validasi input dan business logic. Penelitian ini melakukan eksperimen fuzzing pada Web API dengan pendekatan offensive dan defensive. Penelitian ini membandingkan beberapa state-of-the-art fuzzing tools untuk pendekatan offensive, yaitu EvoMaster, Restler, RestTestGen, Tcases, dan Schemathesis. Penelitian ini juga mengembangkan fuzzing tool baru yang diberi nama OffensiveRezzer. Untuk pendekatan defensive, Penelitian ini menggunakan pustaka validasi input Joi, Zod, Marshmallow, dan Pydantic. Metrik kinerja yang digunakan adalah efektivitas fuzzing tool dalam menemukan bugs/errors dan efektivitas pustaka validasi dalam memvalidasi input data, yang diukur dengan menghitung persentase penurunan error. Hasil evaluasi menunjukkan OffensiveRezzer berhasil menemukan bugs/errors paling banyak dibandingkan dengan fuzzing tools lainnya. Kemudian, masing-masing pustaka validasi memiliki efektivitas sebagai berikut: Joi 97,78%, Zod 96,11%, Marshmallow 97,90%, dan Pydantic 97,90%.
.....Web API technology has been widely used in various application infrastructures because it allows different application services to interact and communicate via network platforms. Web API allows applications to share functionality and data with others, making it a preferred choice for integration across infrastructures. Despite the benefits of Web API, it is not without its security concerns. Many vulnerabilities arise due to misconfigurations or insufficient security mechanisms, which can be prevented by performing functionality testing. One of the critical functionality tests to do is fuzzing. Fuzzing is a testing method to identify vulnerabilities that emerge from flawed input and business logic validations. This research performed fuzzing experiments with offensive and defensive approaches. This research compared several state-of-the-art fuzzing tools for the offensive approach, namely EvoMaster, Restler, RestTestGen, Tcases, and Schemathesis. This research also develops a novel fuzzing tool OffensiveRezzer. For the defensive approach, this research compares several state-of-the-art input validation libraries: Joi, Zod, Marshmallow, and Pydantic. The performance metrics used are the fuzzing tool's effectiveness in finding bugs/errors and the validation library's effectiveness in validating fuzzing payloads, which is measured by calculating the percentage of error reduction. Evaluation results show that OffensiveRezzer found the most bugs/errors compared to other fuzzing tools. Then, each validation library has the following effectiveness: Joi 97.78%, Zod 96.11%, Marshmallow 97.90%, and Pydantic 97.90%.