

INTISARI

Pengembangan Sistem Fuzzing untuk Pengujian Kualitas REST API Menggunakan Multi-Agent Reinforcement Learning dan Semantic Operation Dependency Graph

Oleh

David Lois

22/498373/PA/21509

Pengujian REST API menghadapi tantangan signifikan akibat kompleksitas dependensi antar-operasi dan ruang input yang luas. Alat pengujian *state-of-the-art* seperti AutoRestTest menawarkan solusi efektif melalui *Semantic Operation Dependency Graph* (SODG) dan *Multi-Agent Reinforcement Learning* (MARL), namun penggunaannya terbatas karena antarmuka baris perintah yang kompleks dan minimnya integrasi alur kerja. Penelitian ini mengembangkan sistem *fuzzing* berbasis web yang mengintegrasikan AutoRestTest dengan arsitektur asinkron dan otomatisasi CI/CD. Pengembangan dilakukan secara *end-to-end* mencakup antarmuka manajemen, orkestrasi *task runner*, dan *wrapper* model cerdas. Hasil pengujian performa menunjukkan sistem mampu menangani beban tinggi dengan rata-rata waktu respons 149 ms dan tingkat kesalahan internal 0,01%. Validasi fungsional pada layanan target berhasil mendeteksi 77,8% *broken endpoints* dengan total 16.739 permintaan dalam 60 detik. Evaluasi usability menghasilkan skor akhir 4,86 dari skala 5,00, yang membuktikan bahwa sistem ini efektif menjembatani kesenjangan antara alat riset canggih dengan kebutuhan praktis industri perangkat lunak. Secara nyata, sistem ini berkontribusi pada peningkatan keandalan layanan digital sehari-hari dengan memungkinkan pengembang mendeteksi dan memperbaiki celah keamanan kritis secara dini sebelum berdampak pada pengguna akhir.

Kata Kunci: Fuzzing REST API, Aplikasi Web, Multi-Agent Reinforcement Learning (MARL), Semantic Operation Dependency Graph (SODG), Pengujian Perangkat Lunak Otomatis, Pengujian API Stateful.

ABSTRACT

Fuzzing System Development for REST API Quality Testing Using Multi-Agent Reinforcement Learning and Semantic Operation Dependency Graph

By

David Lois

22/498373/PA/21509

REST API testing faces significant challenges due to complex inter-operation dependencies and a vast input space. State-of-the-art testing tools like AutoRestTest offer effective solutions through Semantic Operation Dependency Graph (SODG) and Multi-Agent Reinforcement Learning (MARL); however, their adoption is often limited by complex command-line interfaces and minimal workflow integration. This research develops a web-based fuzzing system that integrates AutoRestTest with an asynchronous architecture and CI/CD automation. The development is conducted end-to-end, encompassing a management interface, task runner orchestration, and intelligent model wrappers. Performance testing results demonstrate the system's capability to handle high loads with an average response time of 149 ms and an internal error rate of 0.01%. Functional validation on the target service successfully detected 77.8% of broken endpoints with a total of 16,739 requests generated in 60 seconds. Usability evaluation yielded a final score of 4.86 out of 5.00, proving that this system effectively bridges the gap between advanced research tools and the practical needs of the software industry. In practice, this system contributes to the reliability of everyday digital services by enabling developers to detect and fix critical security vulnerabilities early before they impact end-users.

Keywords: REST API Fuzzing, Web Application, Multi-Agent Reinforcement Learning (MARL), Semantic Operation Dependency Graph (SODG), Automated Software Testing, Stateful API Testing.