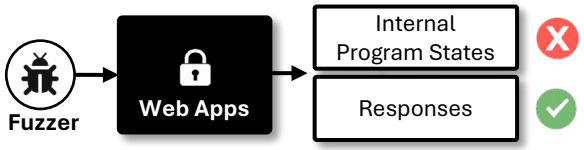
THE ACM WEB CONFERENCE 2026
29 June - 3 July 2026
Dubai, United Arab Emirates
www2026.thewebconf.org
Virtual EVENT

# rfp1898 - Zelda: Feedback-driven Closed-box Fuzzing for Identifying Web Application Vulnerabilities

Soyoung Lee

## MOTIVATION

### Closed-box web fuzzing challenges




- Reaching multi-conditional vulnerabilities
- Designing effective input exploration and exploitation strategies

## KEY IDEAS

1. **Separate exploration and exploitation steps**—non-exploit mutations first for testing coverage, then targeted exploit injection
2. **Coverage inference** from changes in observable HTTP responses
3. **Identify critical input parameters for mutation** by their frequencies and impact on the response

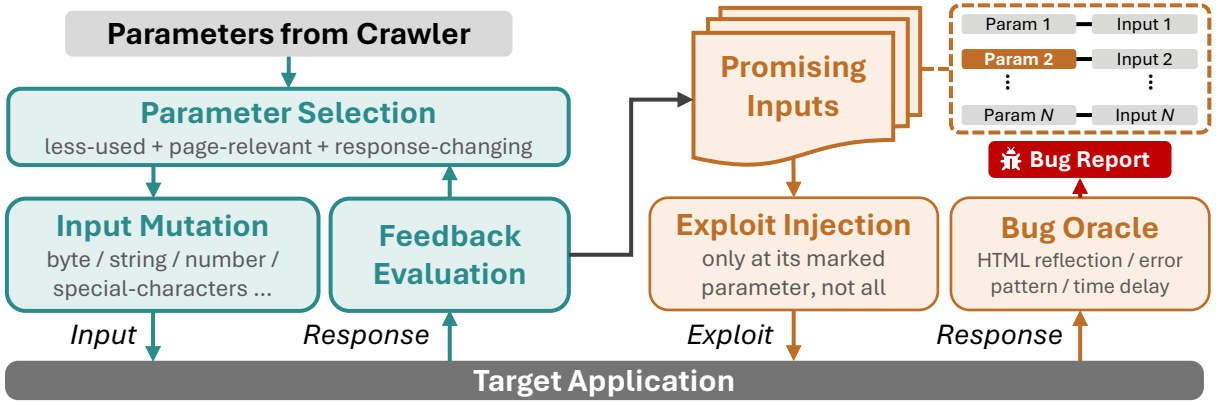
## DESIGN

### Stage 1: Exploration

- Finding promising inputs
- Passing branch/multi-conditional checks

### Stage 2: Exploitation

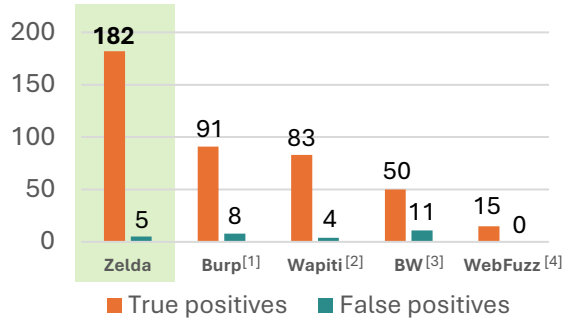
- Confirming vulnerabilities
- XSS, SQL injection, Command injection




## RESULTS

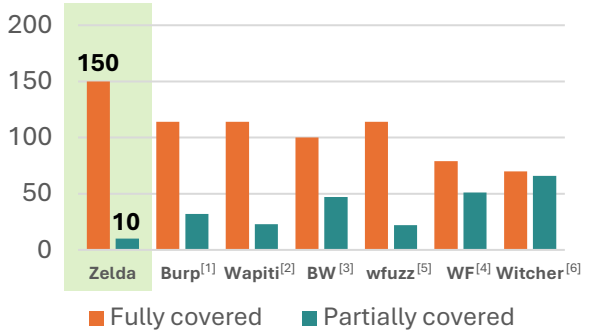
### Vulnerability Detection

- Target: 24 Apps (309 vulnerabilities)

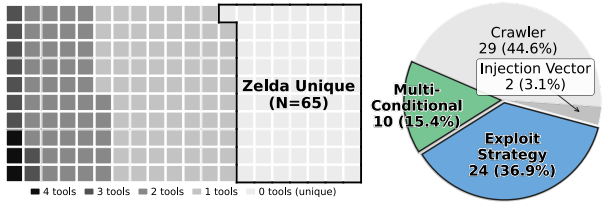



### Path Coverage

- Target: 13 PHP Apps (182 vulnerabilities)
- Did the fuzzer reach the vulnerable sink?




### Unique Vulnerabilities







### Real-world Field Test

- Target: latest versions of benchmark and PHP open-source projects
- **29 CVEs** (8 XSS, 21 SQL injection)

## CONCLUSION

- We propose a **feedback-driven** closed-box web fuzzing framework, Zelda.
- **Separating path exploration** from exploitation contributes to improving vulnerability detection.
- Zelda demonstrates its effectiveness in both **vulnerability detection** and **path coverage**.

## REFERENCES

[1] PortSwigger. Burp Suite. https://portswigger.net/burp.
[2] Nicolas Surribas. Wapiti. https://wapiti.sourceforge.io.
[3] Eriksson et al., Black Widow: Blackbox data-driven web scanning. IEEE S&P 2021.
[4] van Rooij et al., webfuzz: Grey-box fuzzing for web applications. ESORICS 2021.
[5] Xavi Mendez. wfuzz. https://github.com/xmendez/wfuzz.
[6] Trickel et al., Toss a fault to your witcher: Applying grey-box coverage-guided mutational fuzzing to detect sql and command injection vulnerabilities. IEEE S&P 2023.