Universidad
Zaragoza

Ricardo J. Rodríguez <rrodrigu@unizar.es>

Decision on your submission to International Journal of Information Security

International Journal of Information Security <pravin.selvakumar@springernature.com> Wed, Aug 23, 2023 at 4:55 PM
To: rjrodriguez@unizar.es

Ref: Submission ID 96de0921-9cfd-42da-baf9-4b3c2112f796

Dear Dr Rodríguez,

Your manuscript entitled "Automatic Transformation of OpenAPI Specifications to Colored Petri Nets for Security Analysis" has now been assessed. If there are any reviewer comments on your manuscript, please find them below.

Regrettably, the above submission has been rejected for publication in International Journal of Information Security.

Thank you for the opportunity to consider your work. I am sorry that we cannot be more positive on this occasion and hope you will not be deterred from submitting future work to International Journal of Information Security.

Kind regards,

Antonio Lioy
Editor
International Journal of Information Security

Reviewer Comments:

Reviewer 1
BOLA is the top vulnerability in REST APIs according to OWAPS top-10. This paper proposes a novel approach to detect attacks related to this vulnerability based on monitoring a REST API and by matching execution traces with a model extracted from its OpenAPI specification, i.e., a colored petri-net graph. The approach was shown to be effective in a didactic case study.

The presented approach is sound and fully automated. The tool implementation has been exploited in empirical validation. However, the presentation is affected by a few major limitations, that should be addressed before the paper can be accepted for publication:

1. The whole approach is based on a CPN that is constructed following the "link" keyword in the OpenAPI specification of a REST API. However, using this keyword is not very frequent in my experience. Actually, I never saw this keyword used in any specification. Thus, the assumption for the application and adoption of this approach (i.e., the presence "link" keyword) might seem unrealistic. I would recommend authors edit their original manuscript to support the argument that their assumption is realistic. For instance, authors could survey open source REST APIs (e.g., from GitHub) and commercial REST APIs to investigate if (and to what extent) their assumption holds in the real world. Conversely, if the assumption is false, the whole approach losses significance.

2. The paper's title suggests "Security Analysis", i.e., to present an approach to (statically) find vulnerabilities. Instead, the approach is based on monitoring a REST API to detect an attack when it happens, so it is more about attack detection on a deployed service when it is used by the end-users. Please clarify this point and revise the terminology used in the paper.

3. The experimental validation is performed on just a single case study. Moreover, the case study is a didactic software that, as such, probably misses many features of real-world software. To support the external validity of the results, and to make sure that the approach is not biased on this single REST API, I would recommend including more case studies, so as to show that the proposed approach works for real on diversified cases. Considering that it might be challenging to access industrial and commercial software, it could be probably enough to consider open source case studies in Section 5.1, which can be easily found in public software repositories such as GitHub or BitBicket.

Reviewer 2
This paper introduces an approach for identifying the Broken Object Level Authorization (BOLA) vulnerability in RESTful web applications. The process involves two main steps: (1) converting the OpenAPI specification into a colored Petri net (CPN) and (2) evaluating conformance between the CPN and event logs by replaying event traces on the CPN. While step 2 relies on an established method from the literature (reference [5]), the primary focus is on step 1. However, the contribution appears to be limited, and the specific technical problem being addressed is unclear.

(1) The technique for transforming the OpenAPI specification into a colored Petri net (CPN) seems to be relatively straightforward. The provided example with just two transitions and one place appears overly simplistic. A more comprehensive example could encompass more complex scenarios such as choices (if-then-else) and loops.

(2) The paper doesn't effectively communicate why this approach is superior to directly checking event logs against the OpenAPI specification. The rationale for transforming the OpenAPI specification into a CPN is not well elucidated.

(3) The method for BOLA detection by replaying events seems rudimentary in nature.

(4) The absence of related work and lack of evidence highlighting the paper's contribution to a significant research problem are noticeable shortcomings.

Given these factors, it is my recommendation to consider this paper premature for submission to a peer-reviewed publication. I advise rejecting it in its current state.