Evil Pickles: DoS Attacks Based on Object-Graph Engineering (Artifact)*

Jens Dietrich[1], Kamil Jezek[2], Shawn Rasheed[3], Amjed Tahir[4], and Alex Potanin[5]

1 **School of Engineering and Advanced Technology, Massey University**
Palmerston North, New Zealand
j.b.dietrich@massey.ac.nz

2 **NTIS – New Technologies for the Information Society**
Faculty of Applied Sciences, University of West Bohemia
Pilsen, Czech Republic
kjezek@kiv.zcu.cz

3 **School of Engineering and Advanced Technology, Massey University**
Palmerston North, New Zealand
s.rasheed@massey.ac.nz

4 **School of Engineering and Advanced Technology, Massey University**
Palmerston North, New Zealand
a.tahir@massey.ac.nz

5 **School of Engineering and Computer Science**
Victoria University of Wellington, Wellington, New Zealand
alex@ecs.vuw.ac.nz

Abstract

This artefact demonstrates the effects of the serialisation vulnerabilities described in the companion paper. It is composed of three components: scripts, including source code, for Java, Ruby and C# serialisation-vulnerabilities, two case studies that demonstrate attacks based on the vulnerabilities, and a contracts-based mitigation strategy for serialisation-based attacks on Java applications. The artefact allows users to witness how the serialisation-based vulnerabilities result in behavior that can be used in security attacks. It also supports the repeatability of the case study experiments and the benchmark for the mitigation measures proposed in the paper. Instructions for running the tasks are provided along with a description of the artefact setup.

1998 ACM Subject Classification D.2.2 Design Tools and Techniques, D.2.4 Software/Program Verification, D.3.3 Language Constructs and Features, D.3.4 Processors, D.4.6 Security and Protection, E.2 Data Storage Representations

Keywords and phrases serialisation, denial of service, degradation of service, Java, C#, JavaScript, Ruby, vulnerabilities, library design, collection libraries

Digital Object Identifier 10.4230/DARTS.3.2.13

Related Article Jens Dietrich, Kamil Jezek, Shawn Rasheed, Amjed Tahir, and Alex Potanin, "Evil Pickles: DoS Attacks Based on Object-Graph Engineering", in Proceedings of the 31st European Conference on Object-Oriented Programming (ECOOP 2017), LIPIcs, Vol. 74, pp. 10:1–10:32, 2017.
http://dx.doi.org/10.4230/LIPIcs.ECOOP.2017.10

Related Conference European Conference on Object-Oriented Programming (ECOOP 2017), June 18-23, 2017, Barcelona, Spain

* This work was supported by a gift to the first author from Oracle Labs Australia

1 Scope

The artefact is designed to support repeatability of the experiments of the companion paper. It has scripts, including sources, for the security vulnerabilities described in the paper for the programming languages: Java, C# and Ruby, and a case study that demonstrates how the Java vulnerabilities can be used in attacks on two widely used servers, Jenkins deployed on Tomcat and JBoss. It also includes a secure drop-in replacement for `ObjectInputStream` that mitigates attacks based on the described vulnerabilities and a DaCapo-based[1] benchmark to assess the overhead of the instrumentation that implements the contracts-based mitigation strategy.

2 Content

The artefact package includes:

- Script, `~/evilpickles/run-java.sh`, including source code for the Java vulnerabilities (SerialDOS, Pufferfish and Turtles all The Way down) in `~/evilpickles/java`
- Script, `~/evilpickles/run-dotnet.sh`, including source code for the C# vulnerabilities (SerialDOS, Turtles all The Way Down) in `~/evilpickles/dotnet`
- Script, `~/evilpickles/run-ruby.sh`, including source code for the Ruby vulnerability (SerialDOS) in `~/evilpickles/ruby`
- JBoss and Jenkins servers for the case study experiments in `~/evilpickles/case-study`
- Scripts for the case study experiments: `~/evilpickles/case-study/run-jenkins.sh` and `~/evilpickles/case-study/run-jboss.sh`
- Secure replacement for `ObjectInputStream` that uses contracts-based instrumentation to mitigate the attacks in `~/evilpickles/mitigation/dyn` and script to run the experiment: `~/evilpickles/mitigation/dyn/run-mitigation.sh`
- DaCapo-based benchmark for the instrumentation that can be executed with the script, `~/evilpickles/mitigation/dyn/dacapo-benchmark.sh`
- Detailed instructions for using the artefact in `~/evilpickles/instructions.md`

To simplify the setup for the experiments, we provide a VirtualBox disk image containing a Linux distribution fully configured for conducting the experiments. The image contains Ubuntu 16.04.1 server edition along with the source code and the JBoss and Jenkins configured for the case study experiments.

3 Getting the artifact

The artifact endorsed by the Artifact Evaluation Committee is available free of charge on the Dagstuhl Research Online Publication Server (DROPS). The latest version of our code is available on: `https://bitbucket.org/jensdietrich/evilpickles`

4 Tested platforms

The artefact is known to work on any platform running Oracle VirtualBox version 5 (`https://www.virtualbox.org/`). The experiments were conducted on a host machine with 8GB RAM with a processor: Intel(R) Core(TM) i5-6300U CPU @ 2.40GHz

5 License

6 MD5 sum of the artifact

1e7e3686b67256b336dbd24eaa149854

7 Size of the artifact

2.7 GB

References

1 Blackburn et al. *The DaCapo benchmarks: Java benchmarking development and analysis*. Proceedings OOPSLA '06, ACM, 2006