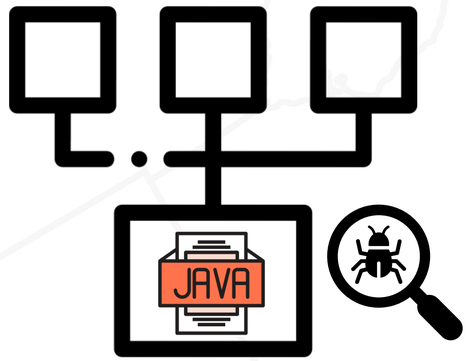
Automatic Security Patching for Containerized Java Applications

Olufogorehan Tunde-Onadele, Xiaohui Gu
North Carolina State University

Motivation

Containerized applications pose a set of new security challenges to distributed computing environments

- An alarming degree of vulnerability exposures exist in official image repositories (Shu et al. 2017)
- Significant resource increase in resource-limited containers can result after patching
- Code analysis schemes have low resource cost but suffer from many false positives
- Many such schemes focus on specific patterns in application and third-party code



Research Questions

- Can code analysis be improved by finding patterns connected to core Java library code?
- Is static analysis practical for security patching containerized applications?

Next Steps

We plan to investigate the following:

- Increase our studied security bugs
- Automatically extract vulnerability patterns
- Generate patching strategies according to pattern categories

Overview

- We have manually investigated **nine** real-world security vulnerabilities in **six** popular web server applications, thus far:
  - Apache ActiveMQ, Apache Commons FileUpload, Apache Tomcat, Apache Struts 2, Elasticsearch, and JBoss

Code Pattern Category	Notable Core Library Functions	Application	CVE	Threat Impact
Unexpected loop conditions	java.io.InputStream.read	Apache Commons FileUpload	CVE-2014-0050	Consume excessive CPU
Invoke functions	javax.xml.transform.Transformer.transform	JBoss	CVE-2015-8103 CVE-2017-12149	Return a shell and execute arbitrary code
Execute functions	org.mvel2.MVELRuntime.execute, java.io.OutputStream.write, com.opensymphony.xwork2.ognl.compileAndExecute	Elasticsearch, Apache ActiveMQ, Apache Struts 2	CVE-2014-3120 CVE-2016-3088 CVE-2017-5638	Return a shell and execute arbitrary code, Execute arbitrary code
Partial condition handling	java.lang.class.forName, java.io.WinNTFileSystem.normalize	Elasticsearch, Apache Tomcat	CVE-2015-1427 CVE-2017-12615 CVE-2020-1938	Return a shell and execute arbitrary code, Disclose credential information

HOTSOS 2020
7TH ANNUAL
HOT TOPICS *in the* **SCIENCE OF SECURITY**
APRIL 7-8, 2020 | LAWRENCE, KANSAS