Why Securing Your Open Source Components Is Critical for AppSec Success

Matt Stanchek
Fortify on Demand Architect
The Use of Open Source

A modern application can include Open Source software as more than 50% of its total codebase
Security Risks from Using Open Source

- Vulnerabilities in Open Source components are inherited by your application
- Many Open Source components use other Open Source components as dependencies
- Vulnerabilities are public
- Exploits are public
Open Source, Open Exploit

Friday, July 9, 2010
CVE-2010-1870: Struts2/XWork remote command execution
Update Tue Jul 13 2010: Added proof of concept

Apache Struts team has announced upload but has not released, due to an unreasonably prolonged voting process. There is a framework which fixes vulnerability that I've reported to them on May 31st 2010. Apache Struts team is ridiculously slow in releasing the fixed version and all my attempts to expedite the process have failed.

Introduction
Struts2 is Struts + WebWork. WebWork in turn uses XWork to invoke actions and call appropriate setters/getters based on HTTP parameter names, which is achieved by treating each HTTP parameter name as an OGNL statement. OGNL (Object Graph Navigation Language) is what turns:

user.address.city=Shihkek&user['favoritesDrink']='kumys'

into
A9:2017-Using Components with Known Vulnerabilities

Components, such as libraries, frameworks, and other software modules, run with the same privileges as the application.

If a vulnerable component is exploited, such an attack can facilitate serious data loss or server takeover.

Applications and APIs using components with known vulnerabilities may undermine application defenses and enable various attacks and impacts.
Legal Risks from Using Open Source

- Some Open Source licenses are more restrictive than others
- Several licenses have “interesting” requirements
- Patent trolls
Demonstration
Wrap Up

- Bill of Materials (BoM)
- Issue transparency
- Seamless experience
Thank You.