Tomcat Native 2

Using our own JNI wrapper for OpenSSL in Tomcat and Undertow

Workshop R&D

Jocelyn Thode & Simon Brulhart

Summary

- Undertow
- Tomcat native
- SSL encryption with Undertow
- Project Goals
- Motivation
- Challenges

Undertow & Tomcat

Undertow and Tomcat are very similar, they:

- Are Web servers
- Are Servlet containers using **JSSE**
- Use SSL **JSSE** Engine

Undertow is used specifically as the Web Server component in **Wildfly**

=> We can use Tomcat Native in Undertow with some modifications

SSL Encryption with Undertow

- Undertow is using **JSSE** SSL Engine for SSL/TLS
- Some work done to use OpenSSL Engine via **JSSE** API (namely sockets)

Tomcat Native

“An optional component for use with Apache Tomcat“

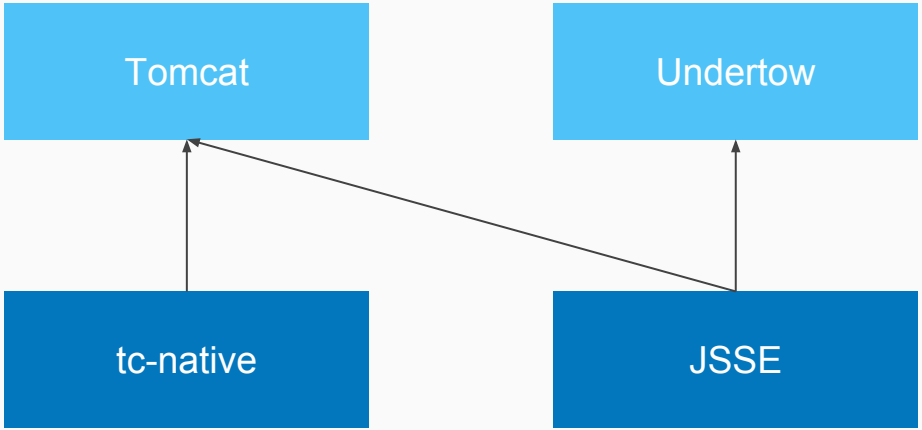
- Better performance and compatibility with OS
- Access to encrypted sockets through Apache Portable Runtime (APR)
- Access to OpenSSL through APR
- Access to OpenSSL through JNI

Project Goals (1)

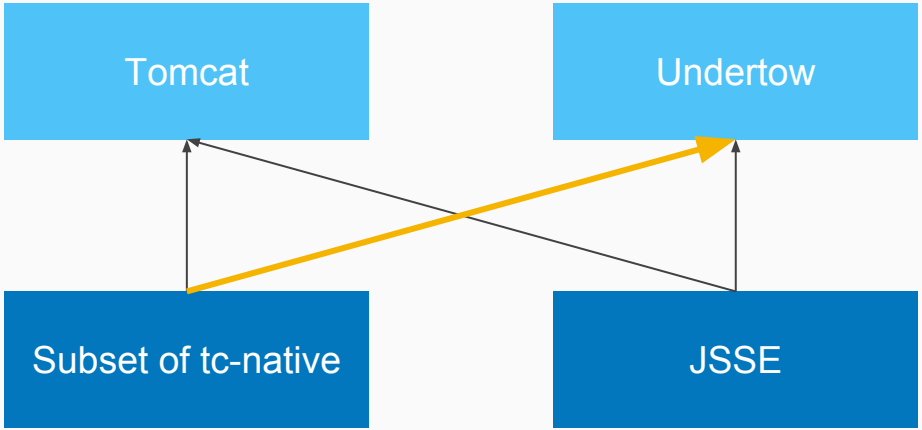
- Study Tomcat Native, Undertow and the current OpenSSL experiments for Undertow
- Subset Tomcat Native so that the JNI interfaces focus on OpenSSL
  - Remove APR completely
- Abstracts Tomcat Native so that it can also be used in **Undertow** and **Tomcat** instead of JSSE
- Run benchmarks to measure performance
- Add JNI calls to access more OpenSSL features (depending on time)

Tomcat Native

Right Now :



Project Goals (2)



Motivations (1)

- Tomcat Native has JNI interfaces for most of the APR APIs
  - We only want a subset that focuses on OpenSSL
    ↳ Easier to maintain, smaller attack surface
- Undertow is using default JSSE engine which performs poorly
- OpenSSL is efficient, active and cross-platform
- Ability to use Tomcat with OpenSSL without loading APR

Using Undertow directly with OpenSSL should perform better but remain easy to maintain

Tentative Schedule

- Study the source code (**~26.03**)
- Remove unneeded code from Tomcat Native (**~10.04**)
- Modify Tomcat Native (**~08.05**)
  - Make it compatible with Undertow
  - Keep compatibility with Tomcat
- Run Benchmarks in Red Hat to test performance (Possibly done by Red Hat) (**~15.05**)
- Implement JNI calls to make more OpenSSL features available in Tomcat Native (**~29.05**)

Challenges

The main challenges of this project will be :

- Understand the different projects and how they work together
  - Red Hat can help
- Adapt Tomcat Native and Undertow to work with OpenSSL natively
  - Numa’s code as well as Stuart’s code can help
- Adapting Tomcat Native to Undertow might not be the best solution
  - Starting from current ssl experiments for Undertow could be easier

Questions