



The Free Javaws Implementation in IcedTea

Omar Majid

Associate Software Engineer

Red Hat

2011-02-05

Overview of the Talk

- Background about JNLP
- Netx's Architecture
- Recent developments in Netx
- Future



Background



What's the name again?

- Java Network Launching Protocol
- Java Web Start

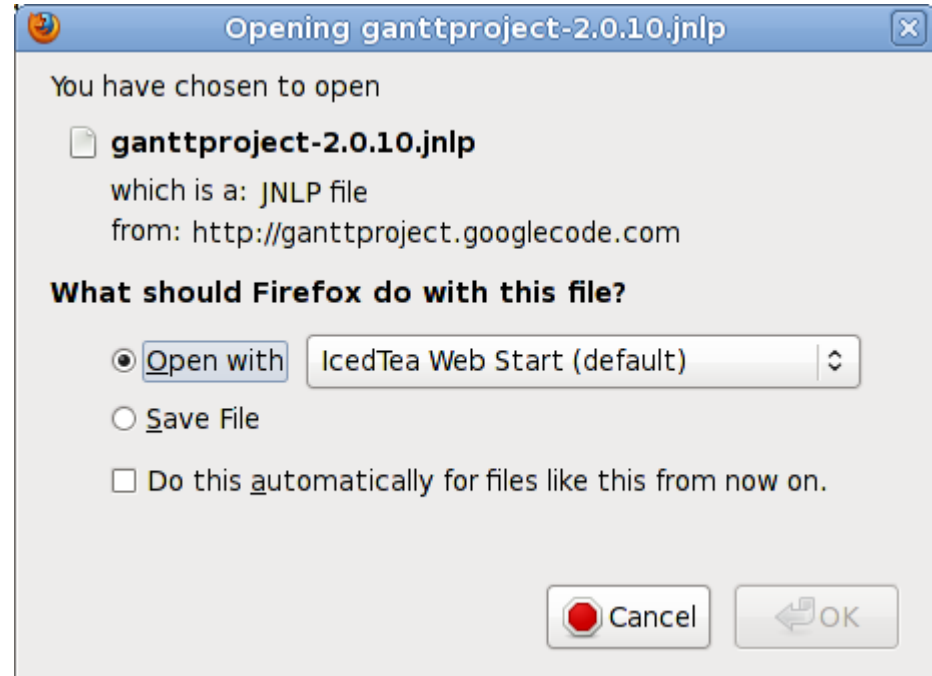
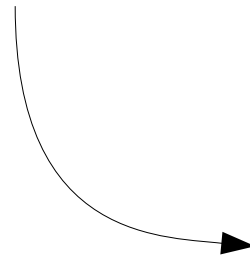


Overview of JNLP

A way to run Java applications
securely without installing them



Running JNLPs - Hyperlinks



Running JNLPs – Command Line

```
$ javaws ganttproject-2.0.10.jnlp
```



The Standard

License Agreement for JSR-000056 Java Network Launching Protocol and API 6.0.18

[Continue »](#) **License Release 3**

To Print: In most browsers, select File and then Print from your browser's menu.

SUN MICROSYSTEMS, INC. ("SUN") IS WILLING TO LICENSE THIS SPECIFICATION TO YOU ONLY UPON THE CONDITION THAT YOU ACCEPT ALL OF THE TERMS CONTAINED IN THIS AGREEMENT. PLEASE READ THE TERMS AND CONDITIONS OF THIS AGREEMENT CAREFULLY. BY DOWNLOADING THIS SPECIFICATION, YOU ACCEPT THE TERMS AND CONDITIONS OF THE AGREEMENT. IF YOU ARE NOT WILLING TO BE BOUND BY IT, SELECT THE "DECLINE" BUTTON AT THE BOTTOM OF THIS PAGE.

Specification: JSR-000056 Java(tm) Network Launching Protocol and API ("Specification")

Version: 6.0.18

Status: Maintenance Release 3

Release: 25 January 2010

Copyright 2010 SUN MICROSYSTEMS, INC.
4150 Network Circle, Santa Clara, California 95054, U.S.A
All rights reserved.

LIMITED LICENSE GRANTS

1. License for Evaluation Purposes. Sun hereby grants you a fully-paid, non-exclusive, non-transferable, worldwide, limited license (without the right to sublicense), under Sun's applicable intellectual property rights to view, download, use and reproduce the Specification only for the purpose of internal evaluation. This includes (i) developing applications intended to run on an implementation of the Specification, provided that such applications do not themselves implement any portion(s) of the Specification, and (ii) discussing the Specification with any third party; and (iii) excerpting brief portions of the Specification in oral or written communications which discuss the Specification provided that such excerpts do not in the aggregate constitute a significant portion of the Specification.
2. License for the Distribution of Compliant Implementations. Sun also grants you a perpetual, non-exclusive, non-transferable, worldwide, fully paid-up, royalty free, limited license (without the right to sublicense) under any applicable copyrights or, subject to the provisions of subsection 4 below, patent rights it may have covering the Specification to create and/or distribute an Independent Implementation of the Specification that: (a) fully implements the Specification including all its required interfaces and functionality; (b) does not modify, subset, superset or otherwise extend the Licensor Name Space, or include any public or protected packages, classes, Java interfaces, fields or methods within the Licensor Name Space other than those required/authorized by the Specification or Specifications being implemented; and (c) passes the Technology Compatibility Kit (including satisfying the requirements of the applicable TCK Users Guide) for such Specification ("Compliant Implementation"). In addition, the foregoing license is expressly conditioned on your not acting outside its scope. No license is granted hereunder for any other purpose (including, for example, modifying the Specification, other than to the extent of your fair use rights, or distributing the Specification to third parties). Also, no right, title, or interest in or to any trademarks, service marks, or trade names of Sun or Sun's licensors is granted hereunder. Java, and Java-related logos, marks and names are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S. and other countries.
3. Pass-through Conditions. You need not include limitations (a)-(c) from the previous paragraph or any other particular "pass through" requirements in any license You grant concerning the use of your Independent Implementation or products derived from it. However, except with respect to Independent Implementations (and products derived from them) that satisfy limitations (a)-(c) from the previous paragraph, You may neither: (a) grant or otherwise pass through to your licensees any licenses under Sun's applicable intellectual property rights; nor (b) authorize your licensees to make any claims concerning their implementation's compliance with the Specification in question.
4. Reciprocity Concerning Patent Licenses.



Other Resources

- Java Deployment Documentation
 - http://download.oracle.com/javase/6/docs/technotes/guides/jweb/deployment_advice.html#deployingApplications
- Existing JNLP Applications



JNLP Application



XML File



Jars



JNLP File Example

```
<?xml version="1.0" encoding="utf-8"?>
<jnlp spec="1.0"
      codebase="."
      href="AddShutdownHook.jnlp">
  <information>
    <title>test adding shutdown hooks</title>
  </information>
  <resources>
    <jar href="AddShutdownHook.jar" main="true" download="eager"/>
  </resources>
  <application-desc main-class="AddShutdownHook"/>
</jnlp>
```



Demo

- Show JNLP File
- Show Code
- Run



Netx



- What Is It?
- Architecture
- Components in Depth
- Netx and the Plugin



Netx

Netx

- Originally at <http://jnlp.sourceforge.net/netx/>
 - Last updated in 2003
- Designed as a library
- About 18k lines of Java



Netx

- Imported into IcedTea in 2008

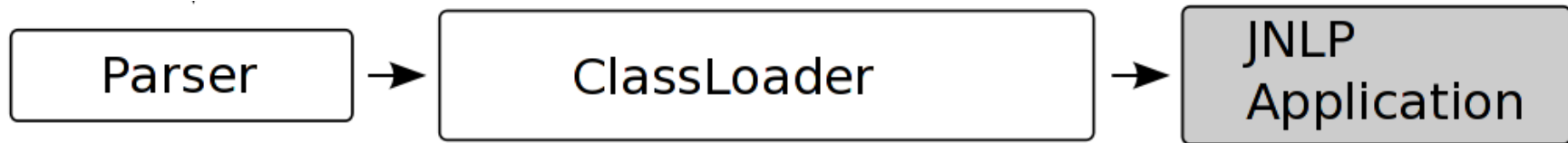
```
changeset: 437:bb4e2512ea29
user:      Joshua Sumali <jsumali@redhat.com>
date:      Wed Jan 09 14:22:14 2008 -0500
files:     ChangeLog Makefile.am Makefile.in README configure configure.ac patches/icedtea-j
description:
2008-01-09 Joshua Sumali <jsumali@redhat.com>
```

```
* Makefile.am: Added building of NetX as our drop-in replacement for
javaws. Since upstream NetX is dormant, we will be hosting and modifying
the NetX source in this repository, particularly in the new tools
directory. The old tools directory which was used in building the
bootstrap/jdk1.7.0 tools.jar during the build has been renamed to
hotspot-tools.
```

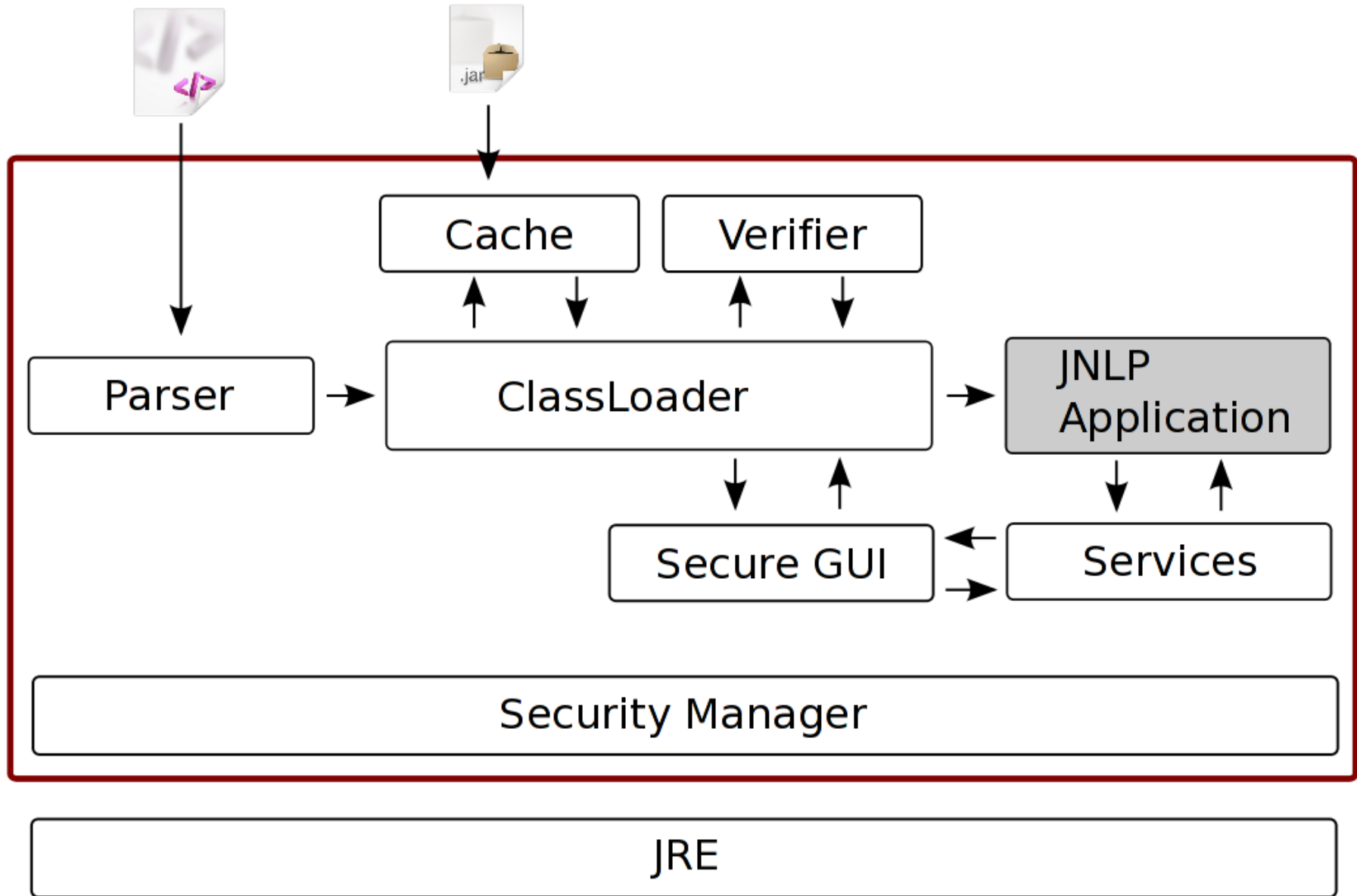
- The default javaws implementation in most distributions
- About 27k lines of code



Overall Architecture



Overall Architecture

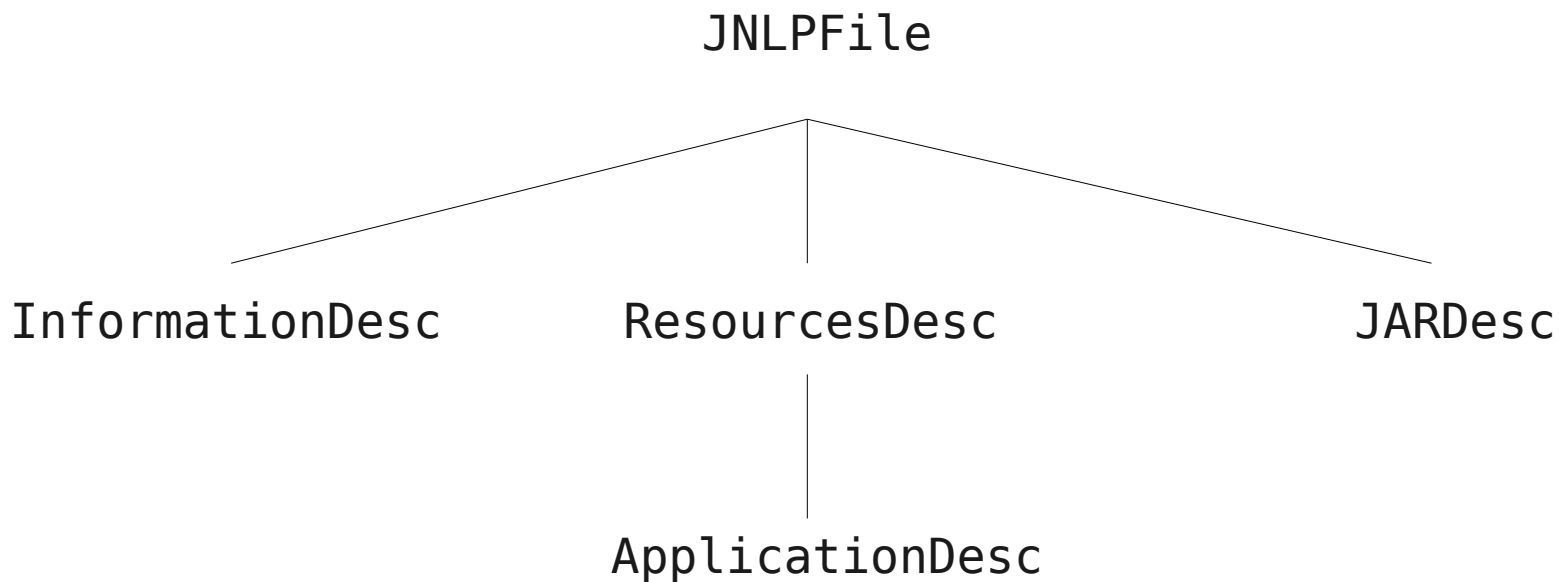


Parser

Parses a JNLP (an XML file) and converts it into a set of java objects



```
<?xml version="1.0" encoding="utf-8"?>
<jnlp spec="1.0"
  codebase="."
  href="AddShutdownHook.jnlp">
  <information>
    <title>test adding shutdown hooks</title>
  </information>
  <resources>
    <jar href="AddShutdownHook.jar" main="true" download="eager"/>
  </resources>
  <application-desc main-class="AddShutdownHook"/>
</jnlp>
```



Parser

- The main class is `net.sourceforge.jnlp.Parser`
- Most of the code is under `net.sourceforge.jnlp`
- Elements are named `<element>Desc`



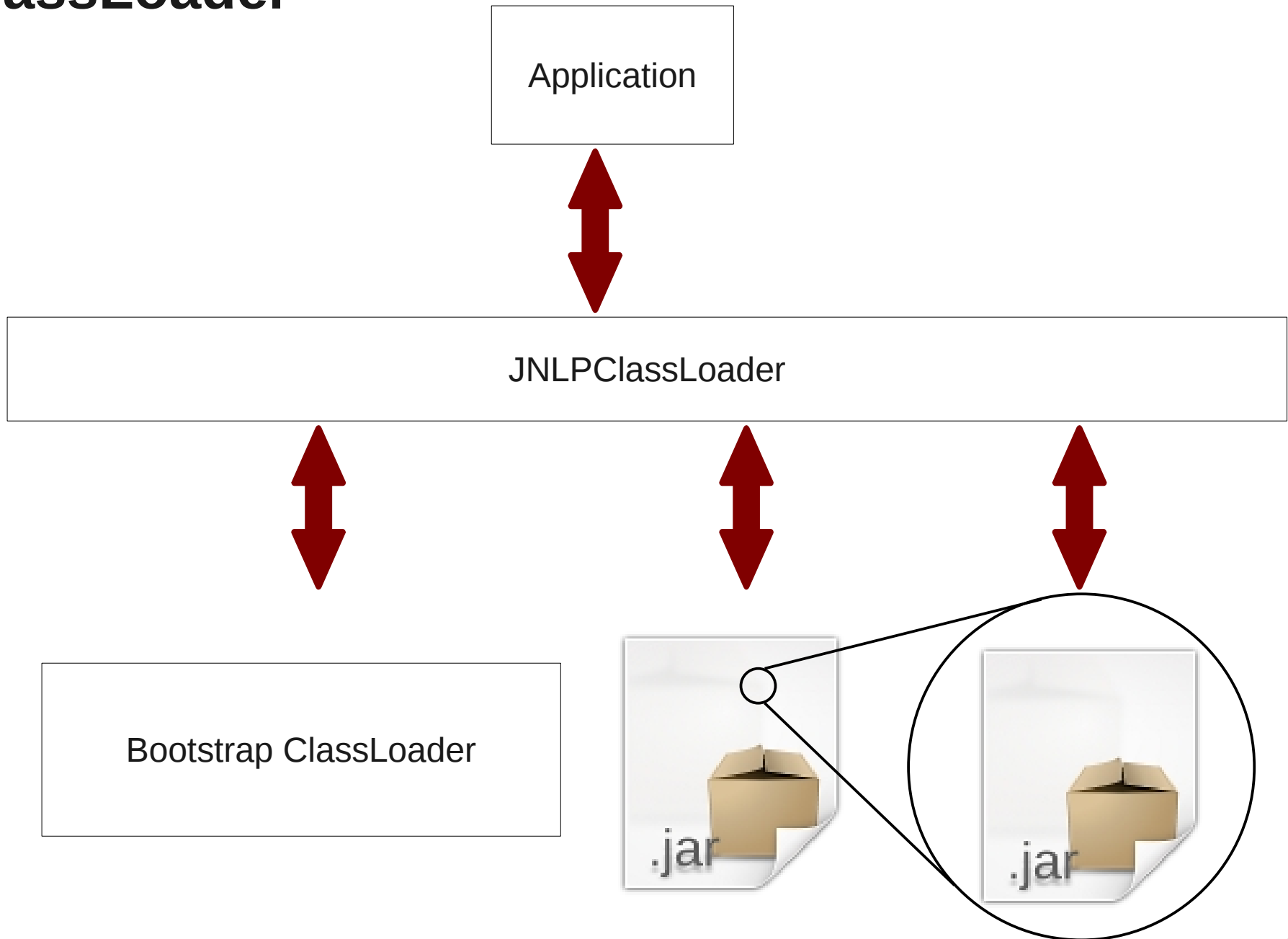
Parser

Uses (in-tree) NanoXML to parse

... and lives with the consequences



ClassLoader



ClassLoaders - Lazy

Download Jars only when needed



ClassLoader - Security

Along with loading classes, associates permissions with them



Cache

- Recall that applications are not installed
- File are cached in `~/.icedtea/cache/`

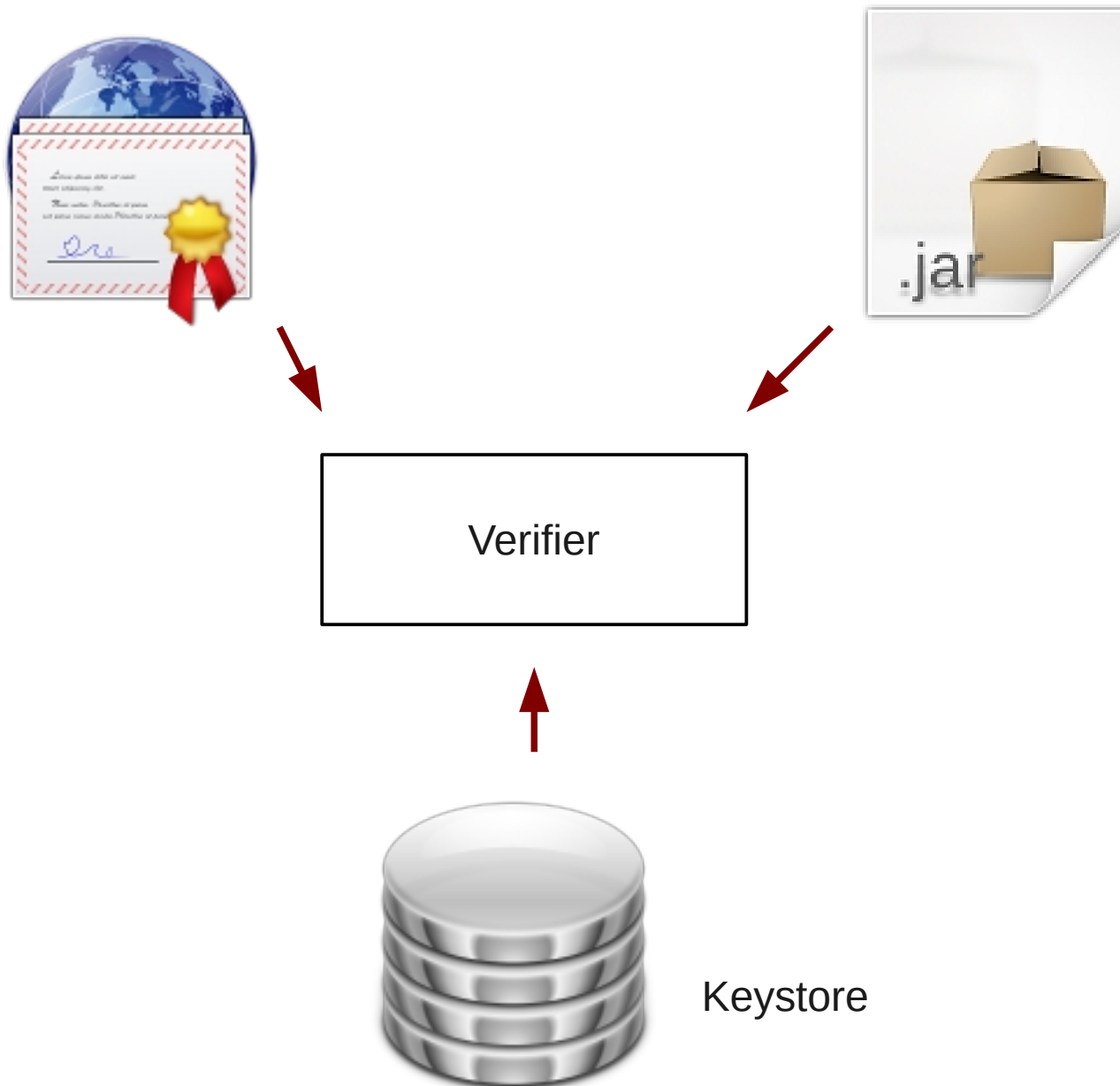


Cache - Downloading

- Multiple threads spawned
- Each file goes through multiple states to arrive at
 - Error
 - Or Downloaded



Verifier



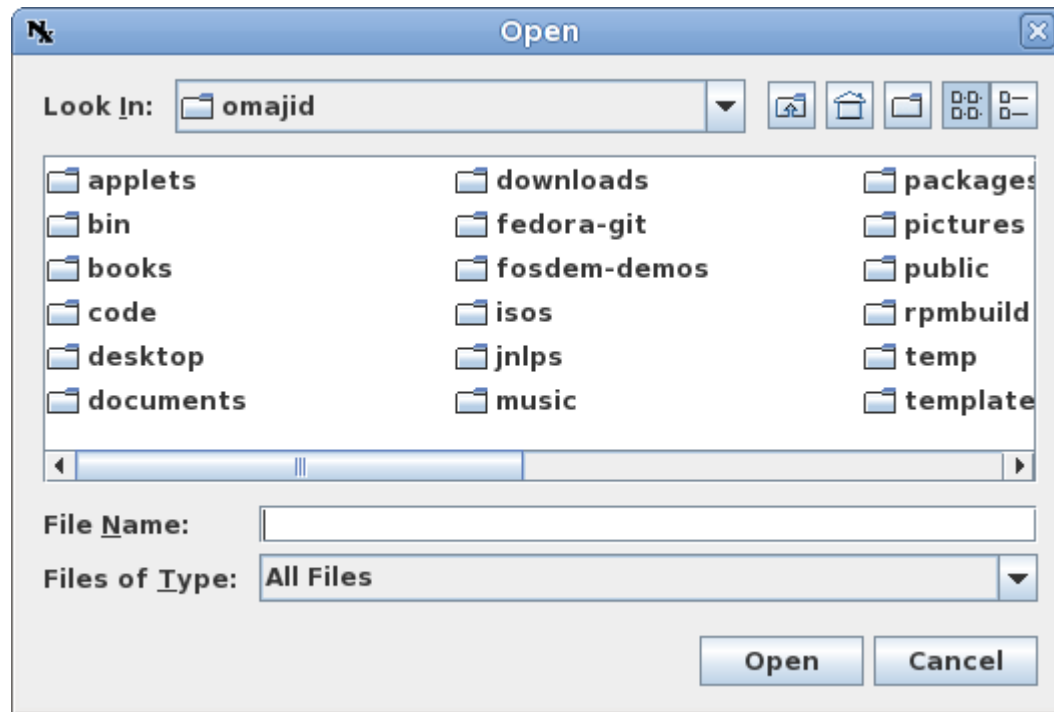
Services

Allow unsigned applications to break the sandbox (securely)



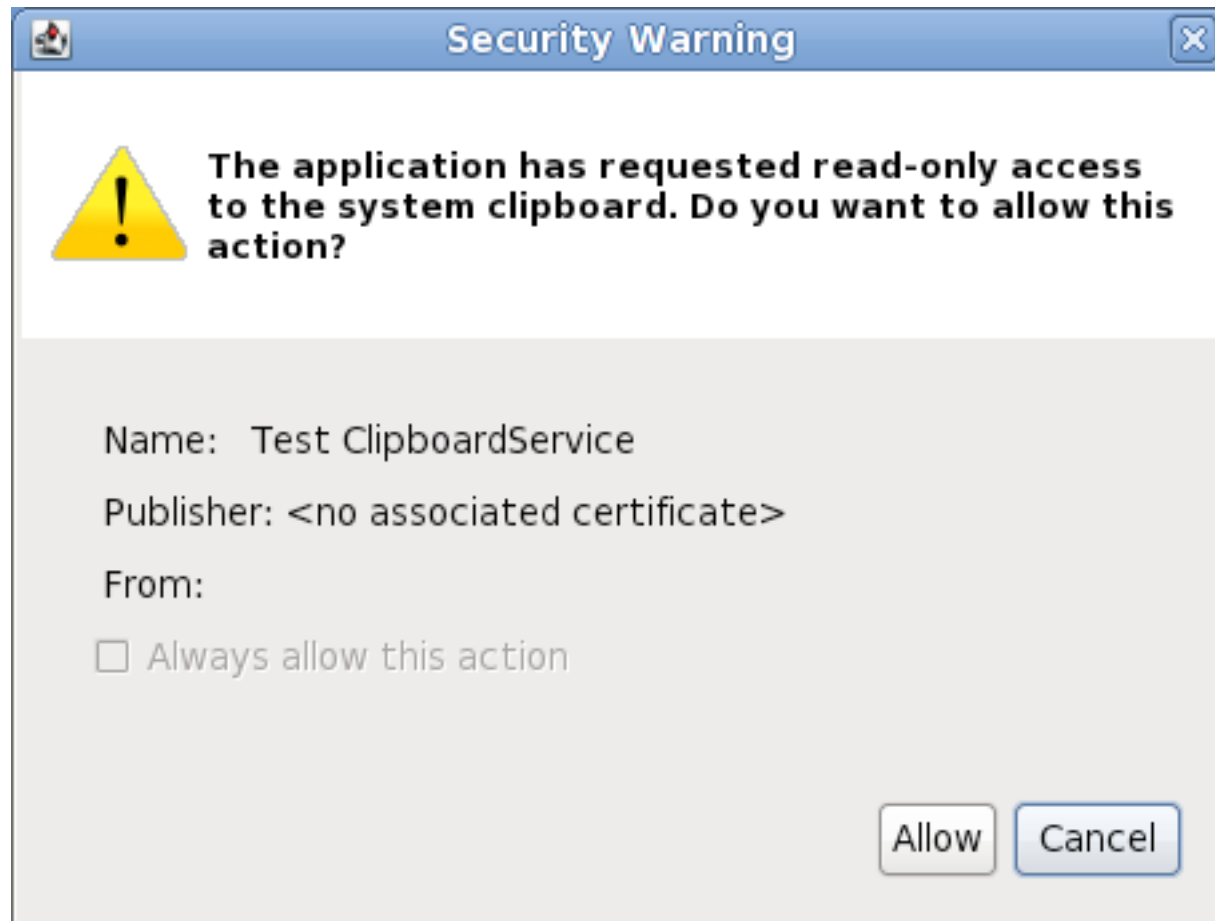
Services

Access to Files

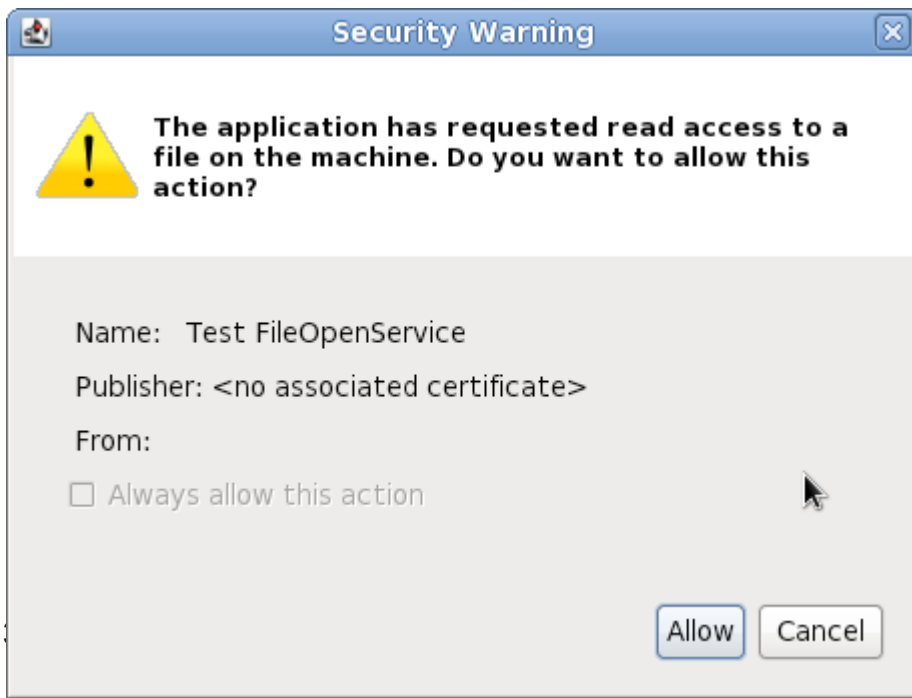
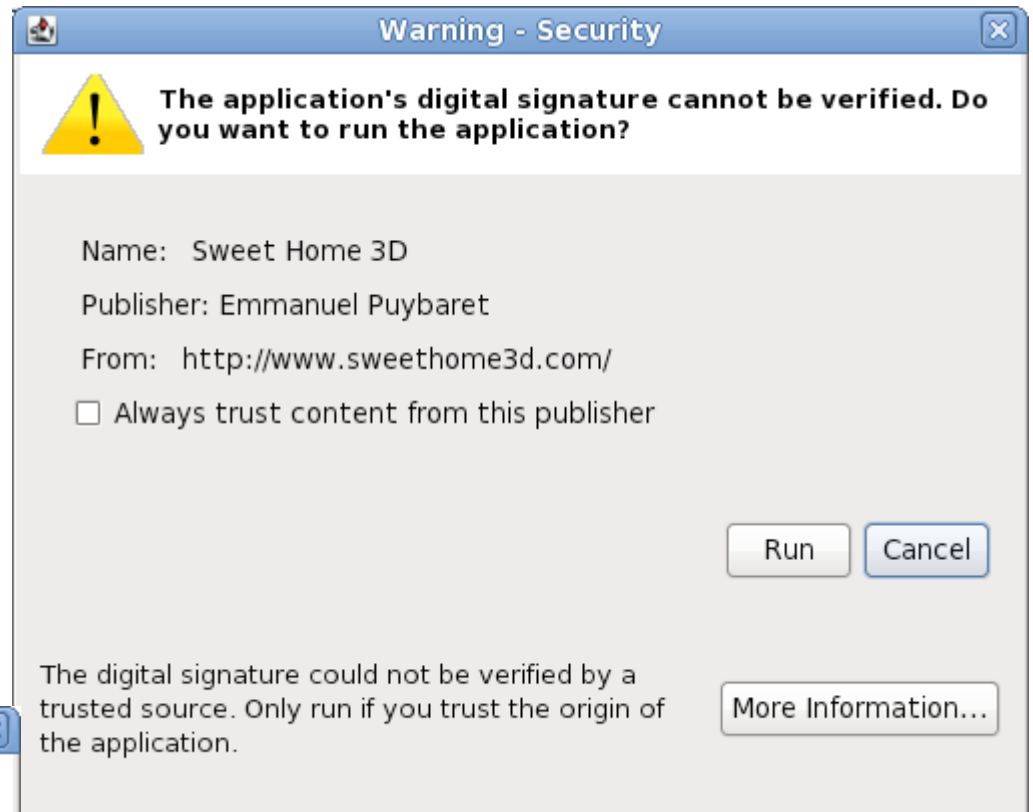


Services - Integration





Secure GUI



Security Manager



- Custom security manager
- Allows some permissions
- Custom policy can be defined



Security/ClassLoader/Verifier Demo



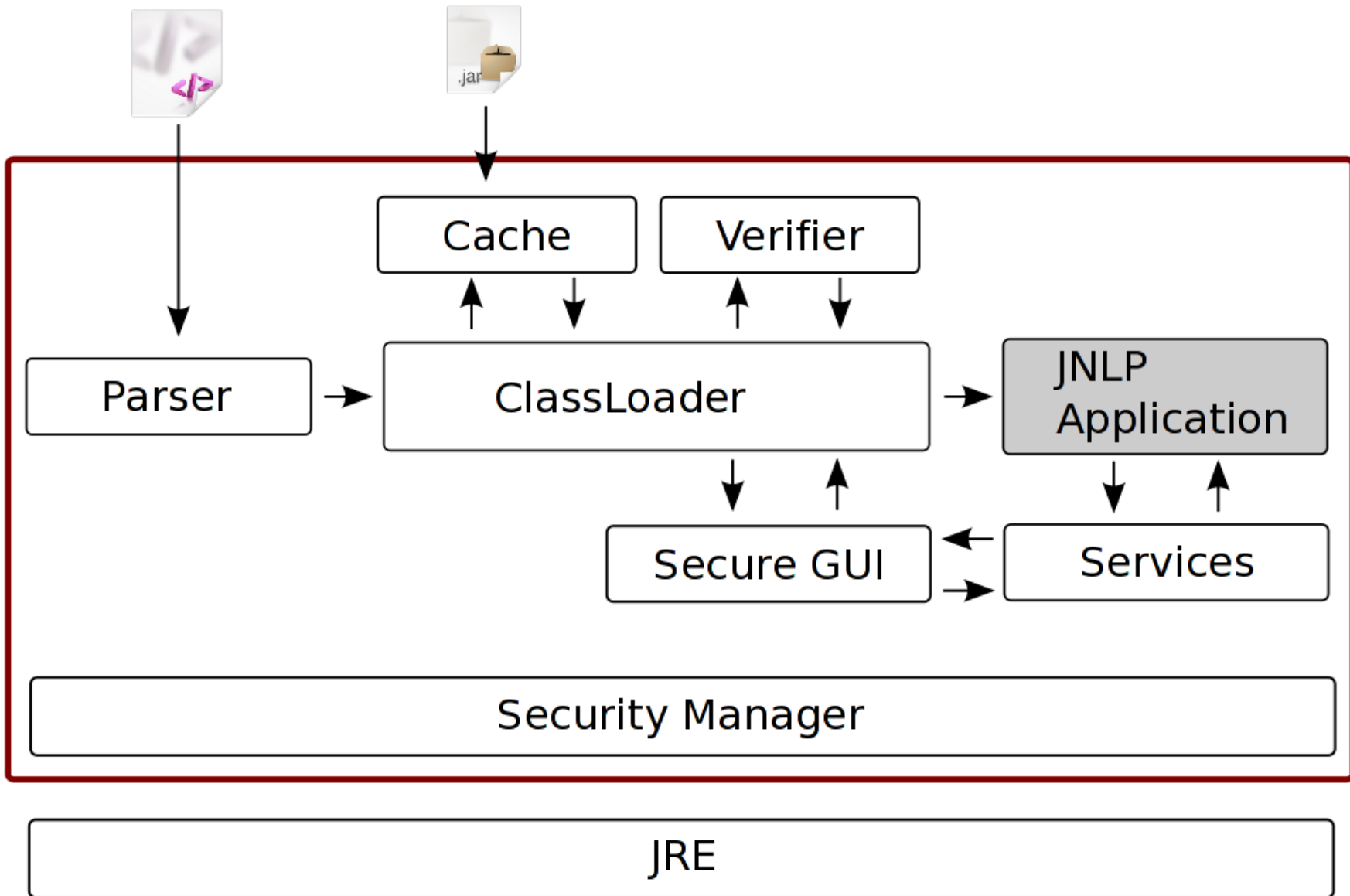
Service Demo



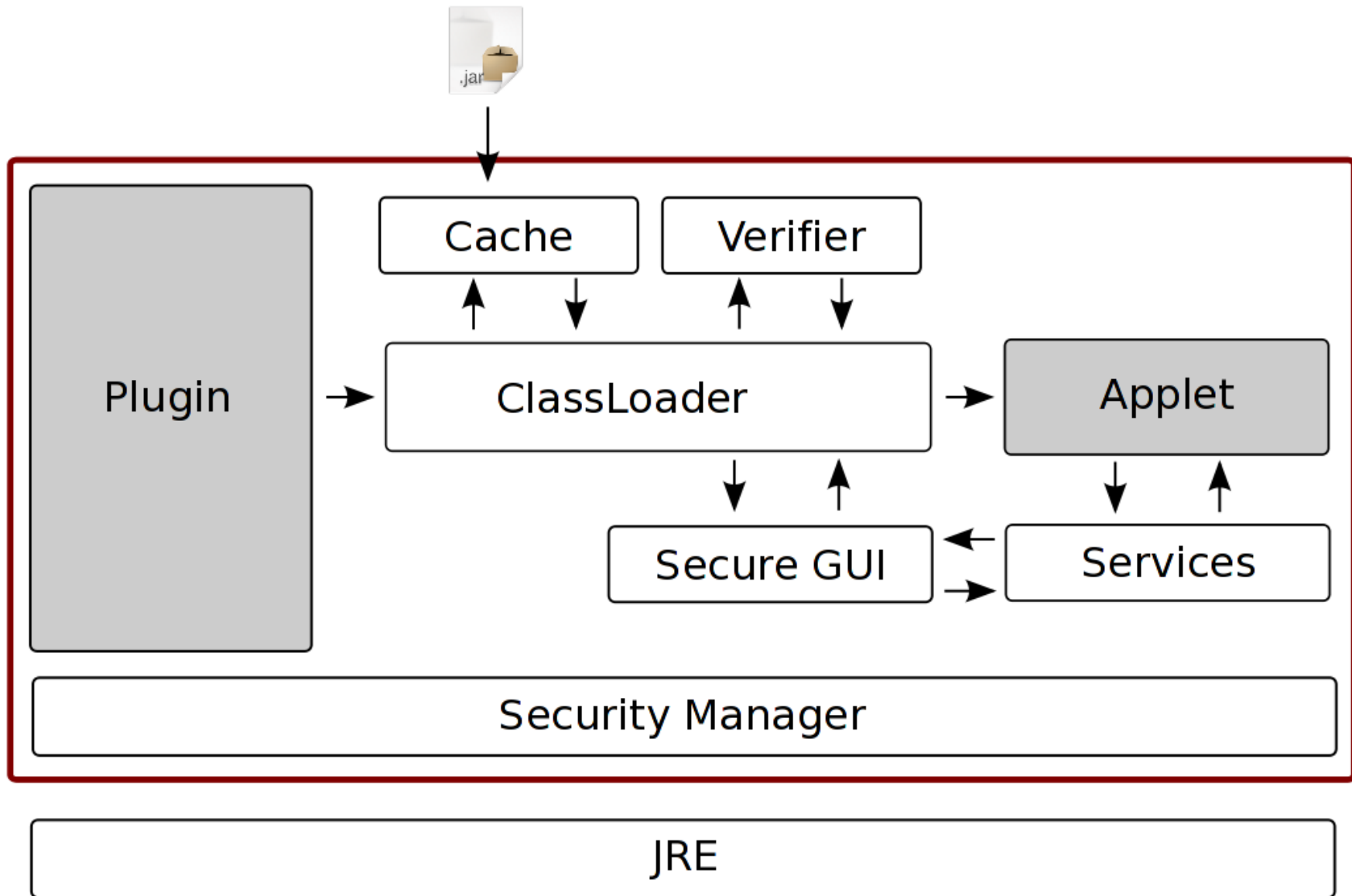
Netx and the Plugin

The plugin relies on Netx for most of its non browser functionality





Plugin in the component framework



Recent Developments in Netx



Versioned/Packed Jars – Sever Side

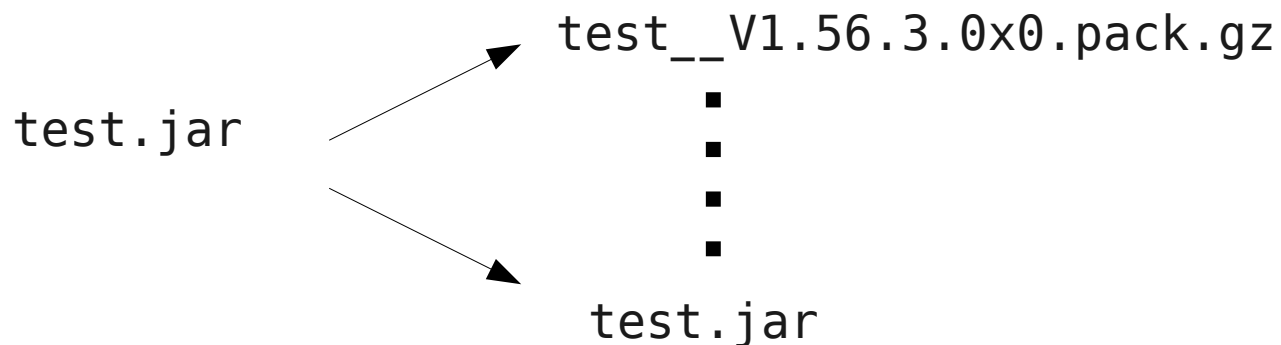
- Server-side
 - content-encoding
 - version-id



Versioned/Packed Jars – Client Side

```
<?xml version="1.0"?>
<jnlp spec="1.0+" codebase="https://server/">
  <information>
    <title>test</title>
  </information>

  <resources>
    <jar href="test.jar" download="eager" main="true" version="1.56.3.0x0"/>
    <property name="jnlp.packEnabled" value="true"/>
    <property name="jnlp.versionEnabled" value="true"/>
  </resources>
</jnlp>
```



Support for deployment.properties

- Allows configuring almost every part of netx
- Described at
 - <http://download.oracle.com/javase/6/docs/technotes/guides/deployment/deployment-guide/contents.html>



Deployment.properties

- Located at `~/.icedtea/deployment.properties`
- Key-value pairs in the standard properties file format



Proxy Support

- Read proxy settings from `deployment.properties`
- Or Firefox's preferences



Custom Policies

- Possible to sandbox trusted applications
- Possible to grant untrusted applications some permissions



TagSoup-based Parser

- Instead of a XML parser, use a html parser

```
<resources>  
  <jar href="test.jar" download="eager"  
</resources>
```



The Future



- Better compatibility
- Implement missing JNLP API services
- Use libproxy
- Better system integration
- Stuff Deepak mentioned...



Questions

