

JSecurityProposal

JSecurity Proposal

Project Name: JSecurity

Introduction

This proposal seeks to create a top-level Apache Software Foundation project to continue the development and advancement of the JSecurity open-source framework. It has broad backing from the JSecurity Community and unanimous backing from the current development team.

We thank you for your consideration.

Key Features & Goals

- The simplest and easiest to understand Java Security API possible.
 - Authentication (log-in) across one or more pluggable *Realms* (JDBC, LDAP, etc), providing PAM (Pluggable Authentication Module) functionality.
 - Authorization (access control) via one or more of said Realms.
 - Dynamic security model support allowing users, roles, and permissions to be changed and assigned dynamically at runtime.
 - Dynamic Instance-level access control - the ability to secure individual instances (files, objects, users, etc) at runtime.
 - POJO-based Enterprise Session Management - access to clustered/distributed/federated user sessions in web *or non-web* environments via the same API.
 - Heterogeneous client session access - access shared session state across client mediums (web MVC, Swing, Flash, C#+SOAP etc).
 - Simple SSO (Single Sign-On) support.
 - Simple Cryptography API.
-

0. Rationale

The current JSecurity community (<http://www.jsecurity.org>) fosters a positive environment of contribution, feedback and supporting fellow members. Although already an open-source project for the last 3+ years, the project as of late has grown quite substantially over the last six months especially, and it is our desire to see JSecurity be adopted by the Apache Software Foundation to continue these efforts. We feel the ASF community will enable the JSecurity project to reach higher adoption rates with better community support beyond what we are able to accomplish ourselves.

Furthermore, a significant number of Apache projects today could find much benefit in JSecurity, as there is not currently anything in the ASF that addresses its feature set as single unified project. We feel that helping other Apache projects would create a symbiotic relationship beneficial to the existing JSecurity community as well as the ASF.

0.1 Criteria

Meritocracy

The JSecurity project will be meritocratic. The project will follow the guidelines (<http://apache.org/foundation/how-it-works.html#meritocracy>) of the Apache Software Foundation. In order to achieve this, we plan on pro actively recruiting individuals in the Community to get involved in the project: specifying work that needs to be done, encouraging bug fixes, enhancements, and advancements, and engaging in discussion on how the code works and is structured. In the end, we are committed to creating an environment to foster a meritocracy.

Community

JSecurity has had a small but active community for its first couple of years after inception, with a significant increase in community members the last 6 months. There are hundreds of posts in the forums, some dating back over 2 years old. Current mailing list activity is around 100+ messages per month and growing with the accumulation of new contributors and users. It is expected that community growth will only continue flourish as an ASF adopted project.

Core Developers

All developers who have ever committed to the existing code repository are still active on the current JSecurity team and will continue to participate. They are:

- Les Hazlewood
- Jeremy Haile
- Tim Veil
- Peter Ledbrook
- Allan Ditzel

Alignment

JSecurity is aligned well with Apache in terms of technologies and licensing. It fits in well technologically with other Apache projects, which also focus on clustering, web frameworks, and Java technologies.

We are sure there are quite a few ASF projects that could find utility in JSecurity. Apache Tomcat might wish to enhance or simplify its Realm functionality by using JSecurity's native support for multiple back-end datasources. There has also already been mention of interest by the Apache Directory [TripleSec](#) team in using JSecurity to support LDAP integration as well as enable dynamic runtime support for security users, roles, and permissions.

Essentially any Apache project that utilizes log-ins, access control, time-based Session access (in both web and non web environments), or cryptography would find JSecurity beneficial.

0.2 Warning Signs

Orphaned products

JSecurity is already a 3+ year old open-source project with a long history and currently in use in many open-source and commercial software products. The interest from the respective communities that use JSecurity (grails.org, et. al.) have continuously grown over the last few years, with very heavy growth in the last 6 months. It is expected that this community growth will only increase with the adoption of the ASF, and the commercial products that use JSecurity will continue to encourage and ensure its longevity.

Inexperience with open source

The current committers have varying degrees of experience contributing and/or committing to open source projects, including Spring (<http://www.springframework.org>), Hibernate (<http://www.hibernate.org>), Grails (<http://www.grails.org>), and others. All have been involved with source code that has been released under an open source license using an open source development process to varying degrees. All current committers are comfortable with normal meritocracy rules, as that is how the development team informally operates now. We do not in any way expect any difficulty in executing under normal ASF meritocracy rules.

Homogeneous developers

All 5 current JSecurity committers work for different companies, with no overlap. They live in different parts of the world, in the United States and Europe.

Reliance on salaried developers

The current committers are not compensated by their employers to contribute to the project. One committer works for G2One (<http://www.g2one.com/>), the company behind the Apache 2.0 open-source Grails platform and *may* contribute to the project on company time. Any such contributions are provided with full knowledge and support of the company, with a valid CCLA on file.

No ties to other Apache products

Currently JSecurity has a required dependency on Apache Commons Logging, with an optional dependency on Apache Commons [BeanUtils](#) Core.

Also based on the above *Alignment* section, JSecurity could very quickly become a part of many other ASF projects, ensuring a successful future within the ASF.

A fascination with the Apache brand

JSecurity started outside of the ASF under the LGPL license. The development team voted unanimously to switch to the Apache 2.0 license to foster a more open community, provide flexible options for commercial deployment, and also to be eligible as an ASF project.

1. Project Scope

The scope of the JSecurity project would be the continued development of JSecurity technology core infrastructure software, including the related utilities and tools. The development would include adding new features and improving performance, scalability, quality, and extensibility.

2. Initial Population Source

The initial resources would be garnered from:

- JSecurity [SourceForge](http://sourceforge.net/projects/jsecurity/) repository (<http://sourceforge.net/projects/jsecurity/>)
-

3. ASF Resources Requested

3.1 Mailing lists

- jsecurity-private (with moderated subscriptions)
- jsecurity-user
- jsecurity-dev
- jsecurity-commits (for SVN commits, automated build notifications, et. al.)

3.2 Revision Control System

JSecurity would like to use a Subversion repository: <https://svn.apache.org/repos/asf/incubator/jsecurity>

3.3 Issue Tracking

Since JSecurity would have its own release cycle, it should have its own JIRA project

- Project Name: JSecurity
 - Project Key: JSEC
-

4. Initial Comitters

- Alan Cabrera (adc@apache.org)
 - Les Hazlewood
 - Jeremy Haile
 - Tim Veil
 - Peter Ledbrook
 - Allan Ditzel
-

5. Sponsoring Individual

We kindly request the Apache Incubator PMC to be the sponsor for this project.

Champion

- Alan D. Cabrera

Mentors

- Alan D. Cabrera
- Paul Fremantle
- Alex Karasulu
- Emmanuel Lecharny
- Craig Russell