Security

On this page:

Fluid Project Security Policy

A closed-membership *Security Group* will be created to oversee all security issues. The group will be headed by a *Security Coordinator* and will include a Board liaison.

Suspected security vulnerabilities should be reported by email to security@fluidproject.org. These messages will be forwarded to members of the Security Group, who are responsible for identifying developers to address the issue.

Notification of security patches or workarounds will first be communicated privately to a vetted *Fluid Adopter List*, which will consist of

- Fluid Project partners
- security mailings lists for the participating projects (Kuali Student, Moodle, Sakai, uPortal)
- key community members
    - community members can request to be added to this list, and will be vetted by the current members of the Security Group

Once members of the Fluid Adopter List have had time to implement the fix, public announcements will be made at an appropriate time, as determined by the Security Group. A mailing list, security-announce@fluidproject.org, will be used for these notifications.

After the mailing list has been notified, information about the fix will be posted on the project website.

For less severe security concerns, patches will be committed to the source code repository with nondescript log messages.

Resources

Mozilla bug reporting practices
Moodle Security Center
JA-SIG Security Contact Group
Sakai Security Documentation

Questions

- We have the ability to apply security levels to JIRA issues, so we could conceivably file JIRA issues for security vulnerabilities, and restrict access to these issues to users of our choice. Do we want to do this?
- Do we want a closed-access wiki space for the Fluid Adopter List members? What would this page be used for?

Developer Security Guidelines

Utilizing AJAX techniques can have tremendous usability benefits for web applications. From a security standpoint, however, AJAX applications have a greater attack surface than traditional web applications. It is important that the use of Fluid components does not open a web application up to increased vulnerability. To that end, the Fluid project provides the following guidelines to help component developers ensure that Fluid components are as secure as possible.

Development Guidelines

1. use Java secure coding practices for any server-side code
2. use Ajax/JS secure coding practices
3. use privacy-related coding practices for sensitive (personally identifiable) information
    - For each of the above, what are the most relevant issues and practices for addressing them?
4. conduct code reviews
    - Who should conduct code reviews? What process should we use?
    - How do we build the process into the community, and how do we maintain it given the resource constraints that we live with?

Testing Guidelines

Ideally, security testing is carried out by people other than the developers themselves. Understanding that this might not be possible, this document attempts to provide guidelines for developers who will be testing their components.

Security testing usually takes the form of manipulating request data to attempt to attack the host. Fluid components are intended to be used by web applications that are outside the control of the component developers, just as a toolkit such as YUI or dojo is intended to be used by any web application.

- What are the implications for security testing?

Developers can use 'insider knowledge' to identify AJAX endpoints.

- An 'endpoint' is a point in the code that is a target for asynchronous calls. This might be an xHttpRequest call in the Javascript, and the format may vary depending on the framework being used (e.g. dojo vs. YUI). Developers can manually review the mark-up and Javascript code to enumerate all known endpoints that may be targets for attack.
- What is the extent of testing we can carry out without knowledge of the server-side?

Resources

JavaScript, DHTML, Ajax and Mashup Security

- Ajax Security Basics
- OWASP Ajax Security Project
- OWASP Ajax Security Guidelines
- IBM Developerworks Shaping the future of secure Ajax mashups
- OpenAjax Whitepaper on Ajax and Mashup Security
- Douglas Crockford's Proposal for the <module> tag
- The Problem with JSON

Best Practices

- Open Web Application Security Project
- OWASP Top Ten
- OWASP Guide to Building Secure Web Applications
- Setting up Subversion Fisheye client

Testing

OWASP Testing Project
OWASP Code Review Project
OWASP Ajax Testing

PAGE HIGHLIGHTS

- Report suspected vulnerabilities to security@fluidproject.org
- Subscribe to security-announce@fluidproject.org to receive public notifications of security patches