Proposal for a User Content and Security Summit

Proposed Format

- 2-day (or 2 half-day) virtual working meeting
- Late April / Early May 2010
- Supported by tele-/video-conference

Objectives

- Understand user goals for markup (e.g., images, styling, video)
- Develop an initial set of technical best practices for supporting the user goals securely
- Develop a plan for evolving the Sakai 2 platform to support and employ the best practices
- Consider how compatible the findings are with Sakai 3

Agenda

See the User Content Summit Agenda page for scheduling and agenda information.

Background

There are two primary forms of textual user content: plain and formatted.

Plain text content should be stored and displayed as entered, without support for formatting or stylistic notation. Plain text inputs are titles, subject lines, and other fields that are used in summary lists, structured headings, or navigational links.

Formatted text content allows the creator to customize the text's appearance and include additional materials like images. This is commonly used in body fields in messages or instructional text. Within Sakai, formatted text uses where HTML markup is accepted and the user is supported by a "Rich Text Editor" such as the FCKeditor. This is discussed here, although other formatting models, such as Wiki text should be considered long-term.

Sakai has a homegrown utility ("FormattedText") for processing each of these types of content as to ensure secure, correctly formatted content to viewers. However, this utility does have some bugs and is not actively maintained. There is also some confusion and lack of clarity around how each tool or service should store and process user content. This has resulted in some expensive maintenance issues in multiple core tools on the 2.6 and 2.7 cycles alone.

The perils of inconsistent practice and an unmaintained library are numerous. Without consistency, it is very difficult to audit source code for appropriate and dangerous techniques, exposing us to an unknown amount of risk across the Cross-Site Scripting (XSS) exploits also emerge over time that may slip past the unmaintained library. Less dangerous, but not less important, is that user demands are evolving in ways that our existing practices cannot accommodate, such as the easy embedding of video from various sources. Institutions are also essentially unable to customize local security policies, which forces the community into a compromise situation.

Research has already been applied to the most technical concern of integrating an actively maintained library to processing. The AntiSamy library from the OWASP project is licensed compatibly and appears to be suitable for quite flexible but does require some policy tuning. This policy should reflect the needs of our users within a generally accepted level of security by default. Institutions could then customize further.

This is the basis of the proposed exercise. Determining the needs of our users, making some recommendations handling of user content, and a technical plan for a transition to a more flexible and supported infrastructure will help the community to understand the resources required and benefits to be gained from undertaking such a project.
The agenda and dates are open for discussion. Please feel free to comment on this page or participate on list. Addressed to Noah Botimer