Kuali Rice 2.3.14 Release Notes

Released: 11-09-2016

Table of Contents

Overview

Welcome to Rice 2.3.14!

Release Highlights

This release consists of a number of security fixes to both the KNS as well as KRAD, including the following:

- Cross-Site Request Forgery (CSRF) protection added to the KNS
- Cross-Site Request Forgery (CSRF) protection added to the KRAD platform
- Fixed a number of XSS vulnerabilities in the KNS and KRAD
- Upgraded commons-fileupload to version 1.3.2

Depending on your use of the KNS and/or KRAD this may be a drop-in replacement. However see below for information on potential impact related to the CSRF implementation.

You will also want to be sure that if you have established dependencies in your own project to commons-fileupload that you upgrade those as well. The specific vulnerability in commons-fileupload is detailed here: http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2016-3092

Download

Kuali Rice 2.3.14 can be downloaded from the Rice website at http://kuali.org/rice/download.

There are three different distributions of Rice available: source, binary and server. Please read the Installation Guide for more details on each of these distributions.

Applications can also consume Rice from the maven site at http://search.maven.org/#search|ga|1|org.kuali.rice.

Documentation

API Documentation can be found at http://site.kuali.org/rice/2.3.14/apidocs/index.html

Formal documentation can be found at http://site.kuali.org/rice/2.3.14/reference/html/index.html. This documentation is still in the process of review and update which will continue through subsequent releases, so please follow the notes in each document to report any outdated information.

Contact

If you encounter any difficulty, please don't hesitate to contact the Rice team on our public collaboration mailing list at `<rice.collab@kuali.org>`. Please indicate that you are using the 2.3.14 version of Rice.

Impacting Changes

CSRF Protection for KNS Applications

This release implements CSRF protection within the KNS. For most standard uses of the framework, no action will be required since the fix for this issue has been incorporated into the KNS's `page.tag` library. However, if you are using the KNS with custom JSP pages or HTML and have custom `<form>` elements then you will need to ensure that you are submitting a hidden form input field that contains the CSRF token.

The CSRF token is stored in the session in an attribute called `csrfSessionToken`. There is a new TAG library in the set of KNS tags called "csrf" which will insert the hidden form field and can be used as follows:

```
<kul:csrf/>
```

This simply resolves to the following HTML:

```
<input type="hidden" name="csrfToken" value="${sessionScope.csrfSessionToken}"/>
```

CSRF Protection for KRAD Applications

This release implements CSRF protection within KRAD. For most standard uses of the framework, no action will be required since the fix for this issue has been incorporated into KRAD's standard `form.ftl` freemarker template. However, if you are using KRAD with custom freemarker pages and creating forms without using the supplied KRAD UIF form component, then you will need to ensure that you are submitting a hidden form input field that contains the CSRF token.

In order to do this, a custom freemarker template has been implemented that will insert the CSRF form field for you. It can be used as follows:

```
<@krad.csrfToken/>
```

This simply resolves to the following:

```
<input type="hidden" name="csrfToken" value="${KualiForm.csrfToken!}"/>
```

Customizing CSRF Protection

If you want to exempt certain paths or entirely disable CSRF protection in your KNS or KRAD application, there are some options available to you.

The following config parameters can be set:

- `csrf.enabled` - enable or disable CSRF protection globally, defaults to “true” if not specified
- `csrf.exempt.paths` - defaults to no paths exemptions, otherwise this is a comma-separated list of partial path matches for which CSRF will not be performed

Alternatively, System Parameters can be created if you need to customize or configure these at runtime. Note that, if defined, the System Parameters will override any config settings. The system parameters are as follows (both use the `KR-SYS` namespace and the `All` component code).

- `CSRF_ENABLED_IND` - as per csrf.enabled
- `CSRF_EXEMPT_PATHS` - as per csrf.exempt.paths