Monitor and organize knowledge objects

As a knowledge manager, you should periodically check up on the knowledge object collections in your Splunk deployment. You should be on the lookout for knowledge objects that:

- Fail to adhere to naming standards
- Are duplicates/redundant
- Are worthy of being shared with wider audiences
- Should be disabled or deleted due to obsolescence or poor design

Regular inspection of the knowledge objects in your system will help you detect anomalies that could become problems later on.

**Note:** This topic assumes that as a knowledge manager you have an *admin* role or a role with an equivalent permission set.

**Example - Keeping tags straight**

Most healthy Splunk deployments end up with a lot of *tags*, which are used to perform searches on clusters of field-value pairings. Over time, however, it’s easy to end up with tags that have similar names but which produce surprisingly dissimilar results. This can lead to considerable confusion and frustration.

Here’s a procedure you can follow for curating tags. It can easily be adapted for other types of knowledge objects handled through Splunk Web.

1. Go to **Settings > Tags > List by tag name**.
2. Look for tags with similar or duplicate names that belong to the same app (or which have been promoted to global availability for all users). For example, you might find a set of tags like `authentication` and `authentications` in the same app, where one tag is linked to an entirely different set of field-value pairs than the other. Alternatively, you may encounter tags with identical names except for the use of capital letters, as in `crash` and `Crash`. Tags are case-sensitive, so Splunk software sees them as two separate knowledge objects. Keep in mind that you may find legitimate tag duplications if you have the *App context* set to *All*, where tags belonging to different apps have the same name. This is often permissible--after all, an `authentication` tag for the Windows app will have to be associated with an entirely different set of field-value pairs than an `authentication` for the UNIX app, for example.
3. Try to disable or delete the duplicate or obsolete tags you find, if your permissions enable you to do so. **However, be aware that there may be objects dependent on it that will be affected.** If the tag is used in reports, dashboard searches, other event types, or transactions, those objects will cease to function once the tag is removed or disabled. This can also happen if the object belongs to one app context, and you attempt to move it to another app context. For more information, see Disable or delete knowledge objects.

4. If you create a replacement tag with a new, more unique name, ensure that it is connected to the same field-value pairs as the tag that you are replacing.

**Using naming conventions to head off object nomenclature issues**

If you set up naming conventions for your knowledge objects early in your Splunk deployment you can avoid some of the thornier object naming issues. For more information, see Develop naming conventions for knowledge objects.