rqst **Admin Guide**

Overview

The rqst app provides new structure and visibility to Splunk project teams, admins, and users. It takes the place of the "tracker" spreadsheet and streamlines data source-specific collaboration. It provides a customizable form inside of Splunk for end-users to submit and monitor requests for data ingest. From an admin standpoint, Splunk engineers can easily filter, view, and interact with these requests, to include adding private notes, tags, and other information. Managers can better understand user needs, resource requirements, and the budgetary impact of individual requests or requests from specific parts of the organization.

All of these capabilities are provided inside of Splunk using only core capabilities, without any external dependencies and with no impact on your Splunk license.

This document provides information relating to the installation and configuration of the rqst Splunk app. If you have any questions, please see the Technical Support section on page 8.

Installation Quick Start

As a user with admin rights, perform the following on your search head:

1. Install rqst through *Apps* > *Install app from file* or manually extract app tarball in your $SPLUNK_HOME/etc/apps directory. Do not restart when prompted.

2. Import the initial app options file into the *rqst_options* KV store collection. (See *KV Store Interaction* below for a recommendation on working with KV store collections.) The initial options file is found here:

   ```
   $SPLUNK_HOME/etc/apps/rqst/appserver/static/setup/rqst_options_initial.csv
   ```

3. Add Splunk Admin Team user information to *rqst_team* KV Store collection. These are the admin users that will interact with user requests. Be sure to flag those with approval authority with `admin_approver = true`.

4. Adjust the *Populate Groups Collection* and *Populate Splunk Users Collection* reports and run them. The groups collection allows for a friendlier mapping of your org to Splunk

role and the users collection exposes a list of users to non-admins.

5. Schedule the *Populate Splunk Users Collection* and the *Update Groups Collection* report to keep the user and group information current.

6. Create a new role *rqst_rest* with Splunk default *power* role inheritance and the following capabilities: *list_storage_passwords, rest_*.* Remove selected indexes from *Indexes searched by default* and *Selected search indexes*.

7. Create a new user *rqst_rest* with the role of *rqst_rest*.

8. Set the *rqst_rest* user password in *passwords.conf* at the command line with:

   ```
   curl -k -u <your_admin_user>:<your_admin_password>
   https://<your_search_head>:8089/servicesNS/nobody/rqst/storage/passwords -d
   name=rqst_rest -d password=<the rqst_rest password set in #7>
   ```

   This will look something like this:

   ```
   curl -k -u admin:changeme
   https://localhost:8089/servicesNS/nobody/rqst/storage/passwords -d
   name=rqst_rest -d password=restchangeme
   ```

   Next, run the following:

   ```
   curl -k -u <your_admin_user>:<your_admin_password>
   https://<your_search_head>:8089/servicesNS/nobody/rqst/storage/passwords/_acl -
   d perms.read=* -d sharing=global
   ```

   Using the same values as our other curl example, this will look something like this:

   ```
   curl -k -u admin:changeme
   https://localhost:8089/servicesNS/nobody/rqst/storage/passwords/_acl -d
   perms.read=* -d sharing=global
   ```

   Please note that these commands are on one line in spite of the document formatting.

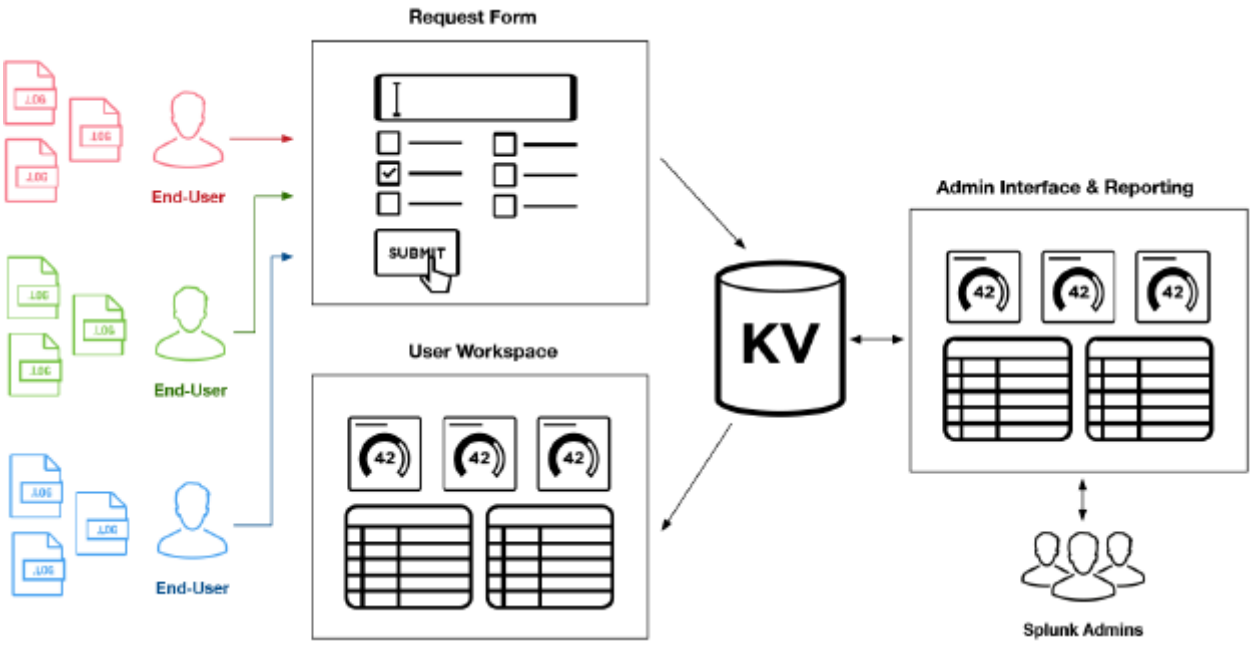
9. Create a new role *rqst_requestor* without role inheritance and add only the *list_storage_passwords* capability or add the *list_storage_passwords* capability to existing user roles. Users that do not have this capability will be unable to submit requests.

10. Submit a test request to confirm proper operation.

11. Schedule the backup of the KV store collections. See *KV Store Backup* below.

KV Store Interaction

The rqst app interacts with the Splunk KV store for all app configuration and data storage/retrieval.



Shameless plug alert: we recommend you take a look at kvkit for KV store administration and interaction. It's the bee's knees. More info is available at https://kvkit.com. Otherwise, we would encourage you to install the Lookup File Editor app (which is also awesome). Both make working with the KV store much easier.

Customization & Configuration Options

Certain aspects of rqst are driven by configurable options. The options shown below are set in the *rqst_options* collection.

`approval_process`

Enable or disable the approval process. If set to "false", new requests will be set to a status of "New" and be immediately available to the admin team. If set to "true", new requests will be set to "Approval" status and require approval by a team member with approval authority before being available to the admin team.

Example value: `true`

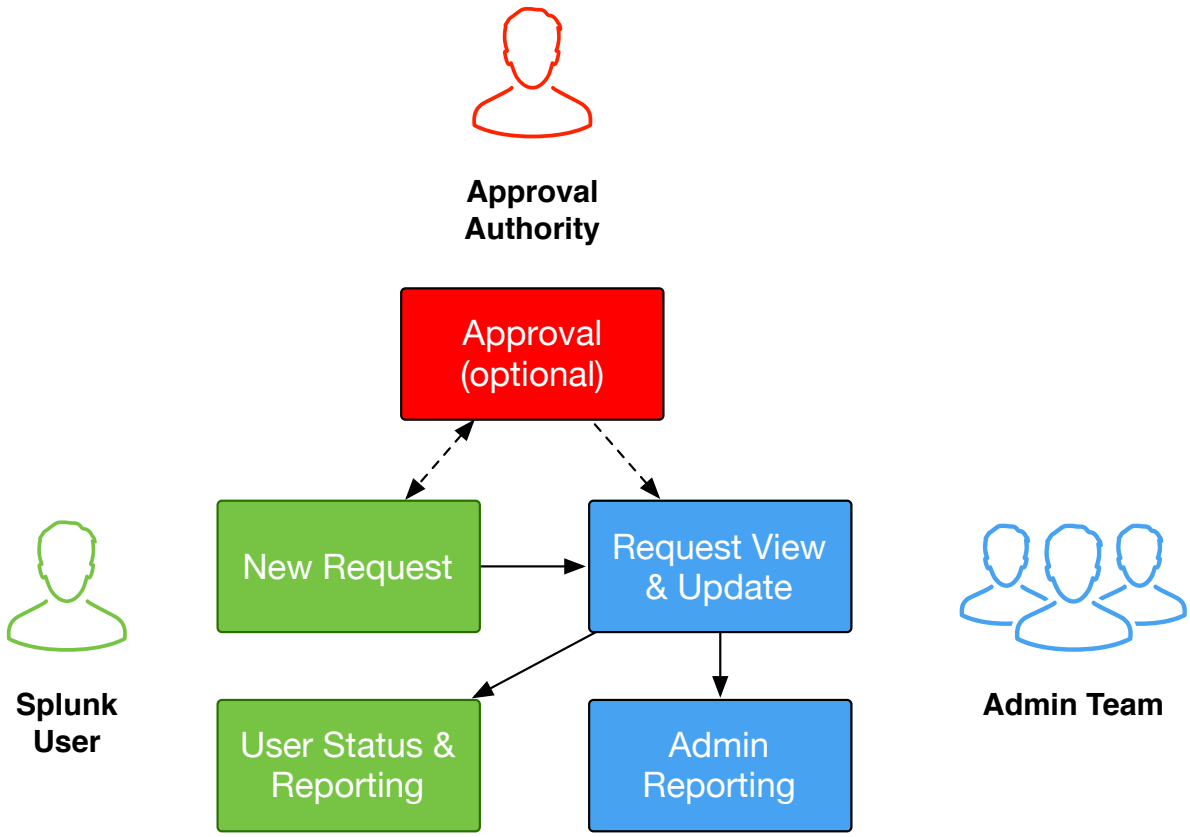


Figure 1.0 – Request Workflow

Interface

`priority`

List of options for the priority field on rqst dashboards.

Example value: `low, medium, high`

`status`

List of options for the status field on rqst dashboards.

Required values: `New, Approval, Hold, Rejected, Working, Complete`
Example value: `New, Approval, Hold, Rejected, Working, Complete, CR Submitted, CR Approved`

`use_case`

List of options for the use case field on rqst dashboards.

Example value: `security, it ops, server, mission, voice`

`data_transport`

List of options for the Data Transport field on rqst dashboards.

Example value: `Universal Forwarder, Heavy Forwarder, Syslog, API`

`indexer_daily_ingest_target`

Your target daily index volume in GB per indexer based on your Splunk environment (hardware, I/O, etc.) and application mix.

Example value: `300`

`cost_per_license_gb`

The cost of license (GB) to be used in dashboard calculations.

Example value: `5.50`

`cost_per_indexer`

The cost of indexer resources to be used in dashboard calculations. Indexer resources, or Indexer Load (IDXL), is determined by requested license `/ indexer_daily_index_target`.

Example value: `15.25`

`cost_per_tb_storage`

The cost of storage (TB) to be used in dashboard calculations.

Example value: `2.50`

`help_url`

The web address that the *Help* button links to in User Workspace. Consider setting this to an internal knowledgebase or intranet site containing information about your Splunk admin team and operations.

Example value: `https://sharepoint.yourcompany.com/something/here/splunk-admin-team`

Email

The rqst app uses Splunk's `sendemail` SPL command to send all email notifications.

`email_notifications`

Enable or disable email notifications. If set to "true", email notifications will be sent to the admin team and users on request creation and update. If set to "false", email notifications will not be sent.

Example value: `false`

`email_server`

Email server used for sending emails when requests are created/updated

Example value: `smtp.gmail.com:587`

`email_new_request_user`

Body of the email sent to the requestor upon request submission.

Example value: `Hey there user, thanks for your request. We're on it!`

`email_new_request_approver`

If the approval process is enabled, the contents of the email sent to team members with approval authority for new requests.

Example value: `Hi approvers! Please do your thing.`

`email_new_request_admin`

If approval process is disabled, the contents of the email sent to admins once a request is created

Example value: `Hey admins! You've got more work to do.`

`email_updated_request_user`

Body of the email sent to a user when their request has been updated

Example value: `Dearest user, your request has been updated! Woo!`

KV Store Collections

The rqst app leverages KV Store collections for all request operations. The table below lists the collections and their role.

Collection Name	Description
rqst_data	Contains main request information.
rqst_audit	Contains log of activity on each request.
rqst_journal	Contains admin notes made on requests.
rqst_groups	Contains mapping of Splunk role to groups, which is used to provide organizational context.
rqst_users	Contains Splunk users and email addresses used to populate the request form.
rqst_team	Contains Splunk admin team members.
rqst_options	Contains options for the rqst app.

KV Store Permissions & Security

In Splunk, a user can have either read or write access to a collection. Giving a non-admin user write access to a collection can have undesirable consequences, such as accidental overwrite. The design of rqst allows non-admin users to write to the *rqst_data* collection via the Request Form while preserving admin-only write permissions.

Contact us if you would like to learn more about this functionality.

KV Store Backup

Since all app data is stored in KV store collections and collections are susceptible to accidental deletion or overwrite (e.g., unintentional `outputlookup` by an admin), **it is highly recommended that you frequently backup all rqst collections to prevent data loss.**

Backup the collections via the CLI:

```
$SPLUNK_HOME/bin/splunk backup kvstore [-archiveName <archive>] [-collectionName <collection>] [-appName <app>]
```

Using the above syntax as a guide, running this command:

```
/opt/splunk/bin/splunk backup kvstore -archiveName rqst_data -collectionName rqst_data -appName rqst
```

will generate a .tar.gz archive in:

```
/opt/splunk/var/lib/splunk/kvstorebackup
```

The archive will contain a JSON file with the contents of the rqst_data collection which could be used to restore.

The command above can be scheduled a la cron. You could create a shell script to streamline the process of backing up each rqst collection. Alternatively, consider using the kvkit application to automatically create backups of kvstore data on a cron schedule. Please see Backup and Restore the KV store in the Admin Manual for official Splunk guidance on the topic.

Technical Support

We're here and happy to help. The free version of rqst (which is what you presumably have downloaded), included base "best-effort" email-only support. We also have premium support options available. Either way, feel free to email support@redfactorapps.zendesk.com.

Appendix A – Open Source Software

The rqst app leverages the software identified in the table below.

Name	URL	License Type & Link
jQuery	https://jquery.com	MIT
Bootstrap	https://getbootstrap.com	MIT
Chosen.js	https://harvesthq.github.io/chosen/	MIT
DOMPurify	https://github.com/cure53/DOMPurify	MPL v2