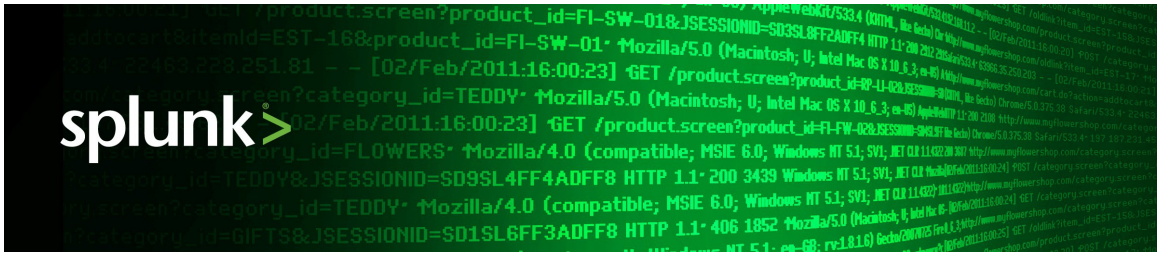

Splunk® Enterprise Knowledge Manager Manual 7.2.5

Configure KV Store lookups

Generated: 4/05/2019 10:14 am

Configure KV Store lookups

KV Store lookups populate your events with fields pulled from your App Key Value Store (KV Store) collections. KV Store lookups can be invoked through REST endpoints or by using the following search commands: `lookup`, `inputlookup`, and `outputlookup`.

This topic shows you how to set up and manage KV Store lookups by configuring lookup stanzas in `transforms.conf`. Configuration files give you a greater degree of control over lookup design and behavior than you get when you set up lookup files using Splunk Web. However, if you do not have access to the `.conf` files, or if you prefer to maintain lookups through Splunk Web whenever possible, you can configure KV Store lookups using the pages at **Settings > Lookups**. See Define a KV Store lookup in Splunk Web.

You can also set up KV Store lookups as automatic lookups. Automatic lookups run in the background at search time and automatically add output fields to events that have the correct match fields. You do not need to invoke automatic lookups with the `lookup` command. See Make your lookup automatic.

Splunk Cloud users: You must use Splunk Web to define lookups. If your Splunk Cloud deployment is a managed deployment, you must request a restart from Splunk Support after uploading lookup files, to make newly uploaded files appear in the list of files available for defining lookups.

For developer-focused KV Store lookup configuration instructions, see Use lookups with KV Store data in the Splunk Developer Portal.

About KV Store collections

Before you create a KV Store lookup, your Splunk deployment must have at least one KV Store collection defined in `collections.conf`. See Use configuration files to create a KV Store collection on the Splunk Developer Portal.

KV Store collections are containers of data similar to a database. They store your data as key/value pairs. When you create a KV Store lookup, the collection should have at least two fields. One of those fields should have a set of values that that match with the values of a field in your event data, so that lookup matching can take place.

When you invoke the lookup in a search with the `lookup` command, you designate a field in your search data to match with the field in your KV Store

collection. When a value of this field in an event matches a value of the designated field in your KV Store collection, the corresponding value(s) for the other field(s) in your KV Store collection can be added to that event.

The KV Store field does not have to have the same name as the field in your events. Each KV Store field can be **multivalued**.

Note: KV Store collections live on the search head, while CSV files are replicated to indexers. If your lookup data changes frequently you may find that KV Store lookups offer better performance than an equivalent CSV lookup.

Define a KV Store lookup stanza in transforms.conf

A `transforms.conf` KV Store lookup stanza provides the location of the KV Store collection that is to be used as a lookup table. It can optionally include field matching rules and rules for time-bounded lookups.

If you want a KV Store lookup to be available globally, add its lookup stanza to the version of `transforms.conf` in `$SPLUNK_HOME/etc/system/local/`. If you want the lookup to be specific to a particular app, add its stanza to the version of `transforms.conf` in `$SPLUNK_HOME/etc/apps/<app_name>/local/`.

Caution: Do not edit configuration files in `$SPLUNK_HOME/etc/system/default`.

The KV Store lookup stanza format

When you add a KV Store lookup stanza to `transforms.conf` it should follow this format.

```
[<lookup_name>]
external_type = kvstore
collection = <string>
case_sensitive_match = <bool>
fields_list = <string>
filter = <string>
```

- `[<lookup_name>]` is the name of the lookup.
- `external_type` should be set to `kvstore` if you are defining a KV store lookup.
- `case_sensitive_match` defaults to true. If set to false, case insensitive matching will be performed for all fields in a lookup table. Output fields and values in the KV Store used for matching must be lower case.

- `collection` is the name of the KV Store collection associated with the lookup.
- `fields_list` is a list of all fields that are supported by the KV Store lookup. The fields must be delimited by a comma followed by a space. A field can be any combination of key and value that you have in your KV store collection.

  By default, each KV Store record has a unique key ID, which is stored in the internal `_key` field. Add `_key` to the list of fields in `fields_list` if you want to be able to modify specific records through your KV Store lookup. You can then specify the key ID value in your lookup operations.

  When you use the `outputlookup` command to write to the KV Store without specifying a key ID, a key ID is generated for you.

- `filter`: Optionally use this attribute to improve search performance when working with significantly large KV Store collections. See Prefilter large KV Store collections.

Configure a KV Store lookup

Prerequisities

- See About lookups for more information on lookups.
- See Make your lookup automatic for information on configuring an automatic KV store lookup.
- See Use configuration files to create a KV Store collection store on the Splunk Developer Portal.
- See Prefilter large KV Store collections for information on prefiltering large KV store collections.
- See Add field matching rules to your lookup configuration for information on field/value matching rules.
- See Configure a time-bounded lookup for information on configuring a time-bounded lookup.

Steps

If you have Splunk Cloud and want to define KV store lookups, file a Support ticket. If you have Splunk Enterprise, perform the following steps.

1. Define a KV Store collection in `collections.conf`.
2. Create a KV Store lookup stanza in `transforms.conf`, following the stanza format described above.

If you want the lookup to be available globally, add its lookup stanza to the version of `transforms.conf` in `$SPLUNK_HOME/etc/system/local/`. If you want the lookup to be specific to a particular app, add its stanza to the version of `transforms.conf` in `$SPLUNK_HOME/etc/apps/<app_name>/local/`.
Caution: Do not edit configuration files in `$SPLUNK_HOME/etc/system/default`.

3. (Optional) Use the `filter` attribute to prefilter significantly large KV Store lookup tables.
   You can speed up lookup searches against significantly large KV Store collections by using the `filter` attribute to restrict the searches.
4. (Optional) Set up field/value matching rules for the KV Store lookup.
5. (Optional) If the KV Store collection contains time fields, make the KV Store lookup time-bounded.
6. (Optional) Make the KV Store lookup an automatic lookup by adding a configuration to `props.conf`.
   If you want the automatic lookup to be available globally, add its lookup stanza to the version of `props.conf` in `$SPLUNK_HOME/etc/system/local/`. If you want the lookup to be specific to a particular app, add its stanza to the version of `props.conf` in `$SPLUNK_HOME/etc/apps/<app_name>/local/`.
   Caution: Do not edit configuration files in `$SPLUNK_HOME/etc/system/default`.
7. Save your `.conf` file changes.
8. Restart Splunk Enterprise to implement your changes.
   If you have set up an automatic lookup, after restart you should see the `output` fields from your lookup table listed in the fields sidebar. From there, you can select the fields to display in each of the matching search results.

Prefilter large KV Store collections

When your KV Store collection is extremely large, performance can suffer when your lookups must search through the entire collection to retrieve matching field values. If you know that you only need results from a subset of records in the lookup table, improve search performance by using the `filter` attribute to filter out all of the records that do not need to be looked at.

The `filter` attribute requires a string containing a search query with Boolean expressions and/or comparison operators (==, !=, >, <, <=, >=, OR , AND, and NOT). This query runs whenever you run a search that invokes this lookup.

For example, if your lookup configuration has `filter = (CustID>500) AND (CustName="P*")`, it tries to retrieve values only from those records in the KV Store collection that have a `CustID` value that greater than 500 and a `CustName` value that begins with the letter P.

Note: If you do not want to install a filter in the lookup definition you can get a similar effect when you use the `where` clause in conjunction with the `inputlookup` command.

KV store lookup example

Here is a KV Store lookup called `employee_info`. It is located in your app's `$SPLUNK_HOME/etc/system/local/` directory.

```
[employee_info]
external_type = kvstore
case_sensitive_match = true
collection = kvstorecoll
fields_list = _key, CustID, CustName, CustStreet, CustCity, CustZip
filter = (CustID>500) AND (CustName="P*")
```

The `employee_info` lookup takes an employee ID in an event and outputs corresponding employee information to that event such as the employee name, street address, city, and zip code. The lookup works with a KV Store collection called `kvstorecoll`. The `filter` restricts the lookup query to records with a customer ID greater than 500 and a customer name that begins with the letter "P".

To see how to make this KV Store lookup automatic by adding a configuration to `props.conf`, see Make your lookup automatic.

Search commands and KV Store lookups

After you save a KV Store lookup stanza and restart Splunk Enterprise, you can interact with the new KV store lookup through search commands.

Use `lookup` to match values in a KV Store collection with field values in the search results and then output corresponding field values to those results. This search uses the `employee_info` lookup defined in the preceding use case example.

```
... | lookup employee_info CustID AS ID OUTPUT CustName AS Name | ...
```

It matches employee id values in `kvstorecoll` with employee id values in your events and outputs the corresponding employee name values to your events.

You can use the `inputlookup` search command to search on the contents of a KV Store collection. See the Search Reference topic on `inputlookup` for examples.

You can use the `outputlookup` search command to write search results from the search pipeline into a KV store collection. See the Search Reference topic on `outputlookup` for examples.

You can also find several examples of KV Store lookup searches in Use lookups with KV Store data in the *Splunk Developer Portal*.