Indexes, indexers, and indexer clusters

Generated: 3/23/2020 12:40 am
Indexes, indexers, and indexer clusters

This manual discusses Splunk Enterprise data repositories and the Splunk Enterprise components that create and manage them.

The index is the repository for Splunk Enterprise data. Splunk Enterprise transforms incoming data into events, which it stores in indexes.

An indexer is a Splunk Enterprise instance that indexes data. For small deployments, a single instance might perform other Splunk Enterprise functions as well, such as data input and search management. In a larger, distributed deployment, however, the functions of data input and search management are allocated to other Splunk Enterprise components. This manual focuses exclusively on the indexing function, in the context of either a single-instance or a distributed deployment.

An indexer cluster is a group of indexers configured to replicate each others' data, so that the system keeps multiple copies of all data. This process is known as index replication, or indexer clustering. By maintaining multiple, identical copies of data, clusters prevent data loss while promoting data availability for searching.

Indexes

As Splunk Enterprise processes incoming data, it adds the data to indexes. Splunk Enterprise ships with several indexes, and you can create additional indexes as needed.

A Splunk Enterprise index contains a variety of files. These files fall into two main categories:

- The raw data in compressed form (rawdata)
- Indexes that point to the raw data (index files, also referred to as tsidx files), plus some metadata files

The files reside in directories organized by age. These directories are called buckets. See How Splunk Enterprise stores indexes.

Splunk Enterprise manages its indexes to facilitate flexible searching and fast data retrieval, eventually archiving them according to a user-configurable schedule. Splunk Enterprise handles everything with flat files; it doesn't require any third-party database software running in the background.

To start indexing, you simply specify the data inputs that you want Splunk Enterprise to index. You can add more inputs at any time, and Splunk Enterprise will begin indexing them as well. See What Splunk Enterprise can index in Getting Data In to learn how to add data inputs.

Splunk Enterprise, by default, puts all user data into a single, preconfigured index. It also employs several other indexes for internal purposes. You can add new indexes and manage existing ones to meet your data requirements. See Manage indexes.

Event processing

During indexing, Splunk Enterprise performs event processing. It processes incoming data to enable fast search and analysis, storing the results in the index as events. While indexing, Splunk Enterprise enhances the data in various ways, including by:

- Separating the datastream into individual, searchable events.
- Creating or identifying timestamps.
• Extracting fields such as host, source, and sourcetype.
• Performing user-defined actions on the incoming data, such as identifying custom fields, masking sensitive data, writing new or modified keys, applying breaking rules for multi-line events, filtering unwanted events, and routing events to specified indexes or servers.

*Getting Data In* describes how to configure event processing to meet the needs of your data. See *Overview of event processing*.

**Index types**

Splunk Enterprise supports two types of indexes:

• Events indexes. Events indexes impose minimal structure and can accommodate any type of data, including metrics data. Events indexes are the default index type.
• Metrics indexes. Metrics indexes use a highly structured format to handle the higher volume and lower latency demands associated with metrics data. Putting metrics data into metrics indexes results in faster performance and less use of index storage, compared to putting the same data into events indexes. For information on the metrics format, see the *Metrics* manual.

There are minimal differences in how indexers process and manage the two index types. Despite its name, event processing occurs in the same sequence for both events and metrics indexes. Metrics data is really just a highly structured kind of event data.

**Indexers**

The indexer is the Splunk Enterprise component that creates and manages indexes. The primary functions of an indexer are:

• Indexing incoming data.
• Searching the indexed data.

In single-machine deployments consisting of just one Splunk Enterprise instance, the indexer also handles the *data input* and *search management* functions. This type of small deployment might handle the needs of a single department in an organization.

For larger-scale needs, indexing is split out from the data input function and sometimes from the search management function as well. In these larger, distributed deployments, the Splunk Enterprise indexer might reside on its own machine and handle only indexing, along with searching of its indexed data. In those cases, other Splunk Enterprise components take over the non-indexing roles. *Forwarders* consume the data, indexers index and search the data, and *search heads* coordinate searches across the set of indexers. Here’s an example of a scaled-out deployment:
For more information on using indexers in a distributed deployment, see Indexers in a distributed deployment.

A Splunk indexer is simply a Splunk Enterprise instance. To learn how to install a Splunk Enterprise instance, read the *Installation Manual*.

**Indexer clusters**

An indexer cluster is a group of Splunk Enterprise nodes that, working in concert, provide a redundant indexing and searching capability. There are three types of nodes in a cluster:

- A single **master node** to manage the cluster. The master node is a specialized type of indexer.
- Several **peer nodes** that handle the indexing function for the cluster, indexing and maintaining multiple copies of the data and running searches across the data.
- One or more search heads to coordinate searches across all the peer nodes.

Indexer clusters feature automatic failover from one peer node to the next. This means that, if one or more peers fail, incoming data continues to get indexed and indexed data continues to be searchable.

The first part of this manual contains configuration and management information relevant for all indexers, independent of whether they are part of a cluster. The second part of this manual, starting with the topic About indexer clusters and index replication, is relevant only for clusters.