Splunk Index Management

Splunk Enterprise stores data as events in indexes.

An event refers to a single data record or log entry. It could be a line from a log file, a message from a network source, or any piece of information that is indexed and processed by Splunk

- Understand splunk indexes
- Buckets Overview
- Custom indexes
- backing up indexes
- Monitoring splunk indexes
- volumes

Index is a repository of data/An index is a specific type of data storage inside Splunk Enterprise/ a directory on a filesystem

$SPLUNK_DB = /opt/splunk/var/lib/splunk/

Two types of indexes in splunk -- event & metrics indexes

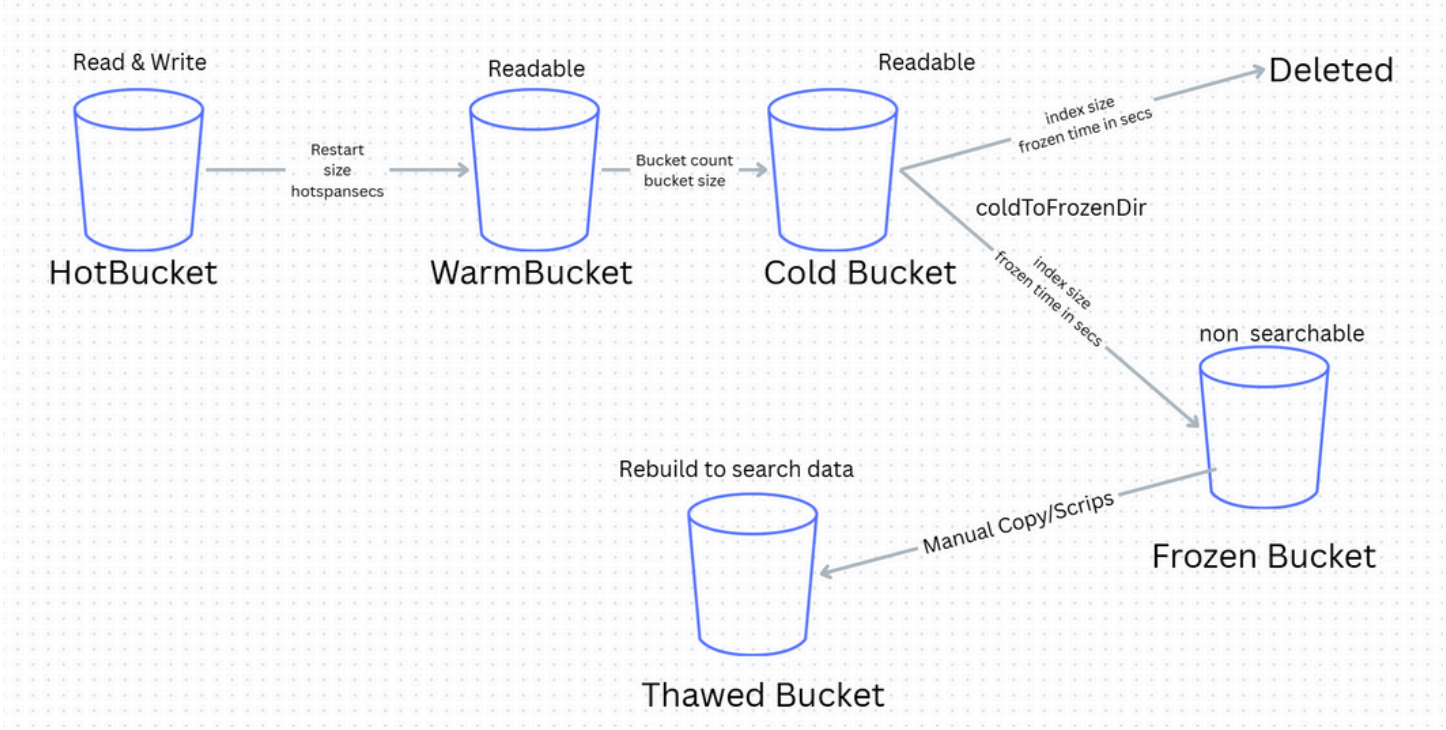
- **Event** indexes store any type of text data, and this is the default index type.
- **Metrics** indexes only store metrics data, which must comply with a defined structure
  - There are special commands in splunk to search metrics indexes data. those are prefixed with m -- **mpreview, msearch, mstats, mcatalog**
  - https://help.splunk.com/en/splunk-enterprise/get-data-in/metrics/9.0/introduction-to-metrics/get-started-with-metrics#Get_started_with_metrics

Default indexes that splunk provides

- **main**: The default Splunk index if one isn't specified in external input sources.
- **summary**: A default summary index for storing summary data. – sourcetype **stash**
- **_internal**: Stores Splunk internal logs from a range of internal sources, such as splunkd daemon logs, web access logs, scheduler logs, python logs and more.
- **_audit**: Stores Splunk internal audit logs such as logins, searches, data accesses, and administrative activities.

- **_introspection**: Stores Splunk resource consumption and performance data. Monitoring console app dashboards highly depend on this index.
- **_thefishbucket**: Stores checkpoint information of file-monitoring inputs. Fishbucket is one which make sure it is not indexing duplicate data for file monitoring inputs. It has checkpoints to track the data monitoring.
- **_telemetry**: Stores instrumentation information if enabled through telemetry.conf.
- lastchanceindex → if specified index is not present in splunk indexer then it will beind landed in lastchangeindex (if configured)

Buckets Overview within an index:



Buckets:

Hot: Hot buckets contain freshly arrived data. Before it is stored, data is parsed in the parsing phase and goes through the license meter. Hot buckets are writable and readable simultaneously. In the index definition, hot buckets are specified to store in the homePath location.

Warm: Hot buckets are rolled over into warm in three scenarios: when the indexer is rebooted,when the hot bucket reaches the maximum size (maxDataSize), or when the maximum lifespan of the bucket is reached (maxHotSpanSecs). Warm buckets are stored in the same location as hot buckets, homePath, and they are only readable. The folder is named using this convention: db_<earliesttime>_<latesttime>_uniqueid. When performing a search in Splunk, the search process determines which buckets to open and search for event matches based on the specified search time range (earliesttime, latesttime).

Cold: Splunk moves warm buckets into coldPath when they reach certain criteria, and bucket names remain the same after they move. The oldest warm buckets are moved first when the homePath location reaches its defined maxDataSizeMB or if the number of warm buckets reaches the defined maxWarmDBCount setting. The data in cold buckets is readable and searchable.

Frozen: Frozen buckets are completely optional and are configured when the administrators choose to retain data from cold buckets. Without frozen bucket configuration, data from cold buckets will be deleted after it reaches a certain age. Frozen buckets are not searchable. One can specify either coldToFrozenScript or coldToFrozenDir to move raw data from a cold bucket to a frozen bucket. If neither of these options is specified, then the data in the cold bucket will be permanently deleted. If you specify both options, coldToFrozenDir takes precedence and coldToFrozenScript is ignored. The criteria for deleting or moving data from a cold bucket to a frozen bucket is either when the data in the cold bucket reaches the age specified in the frozenTimePeriodInSecs option or when the index's maximum total data size is reached. The age of the bucket takes precedence over the size of the index when moving or deleting data from a cold bucket.

Thawed: This is also an optional bucket and it works in coordination with the Frozen bucket. We have seen that frozen buckets are not searchable. To make data in frozen buckets searchable, data from frozen buckets must be copied to thawedPath and rebuilt using the **./splunk rebuild <bucket>** command. Data can be retained in thawedPath as long as it needs to be searched, and no retention policies apply. This means it will never be deleted by Splunk Enterprise. If the data is no longer required for searching, then the administrator can remove it from thawedPath.

Custom indexes can be create in 3ways in most of the cases.

- **UI** -- settings --> indexes --> new index
- **CLI** -- splunk add index <index_name> -<parameter> <value>
  - Parameters: homePath, coldPath, thawedPath, app, and so on
  - splunk add index os -app search
- **Conf**

indexes.conf

[os]

coldPath = $SPLUNK_DB/os/colddb → **cold db buckets**

enableDataIntegrityControl = 0

enableTsidxReduction = 0

homePath = $SPLUNK_DB/os/db → **hot & warm buckets**

maxTotalDataSizeMB = 500 --> **Total index size limit**

thawedPath = $SPLUNK_DB/os/thaweddb

`frozenTimePeriodInSecs =`

There are ways to control the size of data from the bucket directory level to the total index size. Let’s

take a look at the following advanced options and what to back up:

• homePath.maxDataSizeMB limits the homePath directory size for hot and warm buckets.

• coldPath.maxDataSizeMB limits the coldPath directory size for cold buckets.

maxHotBuckets = <positive integer> | auto

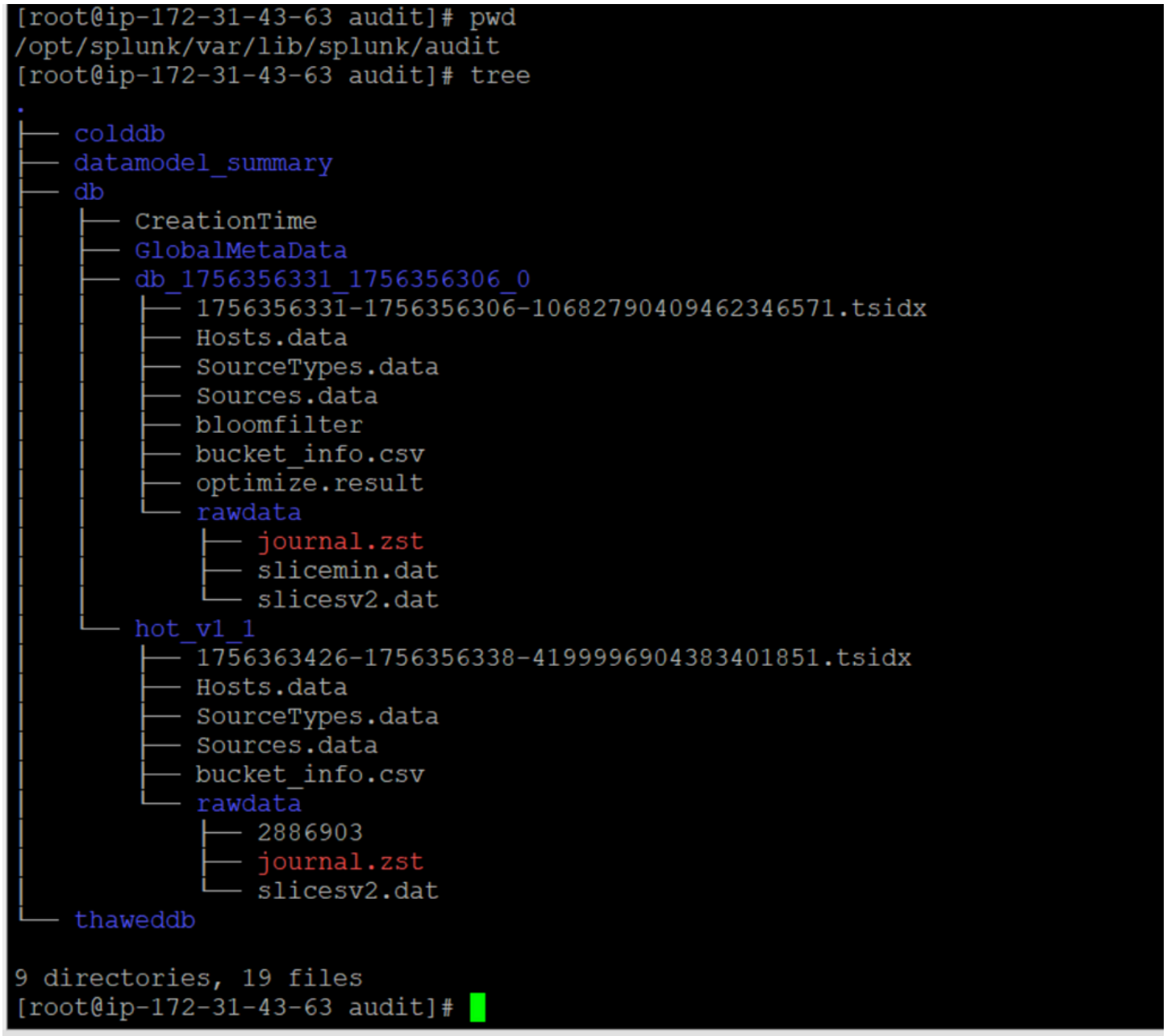
maxWarmDBCount = <nonnegative integer> --> defautl value 300

• To limit the number of hot and warm buckets that can be created, set maxHotBuckets and maxWarmDBCount, respectively. For high-volume indexes, where the indexing size is more than 10 GB/day, Splunk suggests increasing maxHotBuckets from the default of 3 to 10.

maxDataSize = <positive integer>|auto|auto_high_volume

• By default, Splunk limits the hot bucket size to 750 MB through maxDataSize = auto, and if the index size per day estimation is 10 GB, set this option to maxDataSize =

auto_high_volume for Splunk to create 10 GB hot buckets on 64-bit systems. The reason for having this option is to avoid creating too many small buckets for large indexes. Having too many small buckets affects Splunk’s performance. Hot buckets will roll to warm buckets once they reach their maximum size and then a new hot bucket will be created.



Volumes:

[volume:hot_storage]

path = /mnt/fast_disk

Optional limits the volume size to 60 GB

maxVolumeDataSizeMB = 61440

#This setting limits the total size of all databases that reside on this volume to the maximum size specified, in MB

[volume:cold_storage]

path = /mnt/slow_disk

#Optional limits the volume size to 50 GB

maxVolumeDataSizeMB = 500

#index definition

[os]

homePath = volume:hot_storage/os

coldPath = volume:cold_storage/os

maxTotalDataSizeMB = 51200

thawedPath = $SPLUNK_DB\os\thaweddb

thawedPath must be specified, and cannot use volume: syntax

choose a location convenient for reconstitition from archive goals

For many sites, this may never be used.

thawedPath = $SPLUNK_DB/idx1/thaweddb

Backing up indexes:

During hardware failure/DR

snapshot of $SPLUNK_DB (/opt/splunk/var/lib/splunk/)

delete command/ clean command

delete --> to execute delete command one should have can_delete capability assigned. This applies delete markers to data without removing data from storage. Once executed will not be available for search.

To remove data permanently from disk/storage. execute below command on CLI.

--> splunk clean eventdata -index <index_name>

These are to be executed with extreme cautionm, as once deleted cannot be retrieved.