

Splunk Deletion Detector

When one search won't do

Ellie Baum

Lead Software Engineer | Salesforce



Forward-Looking Statements



During the course of this presentation, we may make forward-looking statements regarding future events or plans of the company. We caution you that such statements reflect our current expectations and estimates based on factors currently known to us and that actual events or results may differ materially. The forward-looking statements made in the this presentation are being made as of the time and date of its live presentation. If reviewed after its live presentation, it may not contain current or accurate information. We do not assume any obligation to update any forward-looking statements made herein.

In addition, any information about our roadmap outlines our general product direction and is subject to change at any time without notice. It is for informational purposes only, and shall not be incorporated into any contract or other commitment. Splunk undertakes no obligation either to develop the features or functionalities described or to include any such feature or functionality in a future release.

Splunk, Splunk>, Data-to-Everything, D2E and Turn Data Into Doing are trademarks and registered trademarks of Splunk Inc. in the United States and other countries. All other brand names, product names or trademarks belong to their respective owners. © 2020 Splunk Inc. All rights reserved

Ellie Baum

Lead Software Engineer (PO and recovering SM) | Salesforce



Marketing Cloud CI Team

Everything from Developer desktops to CI builds

- Developer desktop productivity tooling
- Local and CI builds
- All products get telemetry built in

**Without fail, adding data always finds bugs.
More data = More feedback. Less bugs.**

Problem:
My company was deleting
data out of my team's
Splunk index and nobody
knew why





index=ci

How is the build doing?

Be Agile. Get feedback. Iterate.

Data = Feedback

- Contains all our Continuous Integration (CI) data.
- Be a good scientist and hold some variables constant.

**Within days of first having this data,
we found a bug in our CI pipeline that
cut 25% of time off of local dev builds**

Losing This Data Is Not an Option.

Why Can't I Use a "Regular" Search?

This alert only mattered in
context.

Example: If the number of days in
the index was 30 today, I only care if
the number of days in the index
yesterday was 40!

End Goal



Slack Notification

ci-notify

For notifications not code related.

Monday, June 8th

email APP 6:20 PM

From [redacted]
To [redacted] email
Subject: **Splunk Alert: McciDataDeletionDetector**
Date: Jun 8th

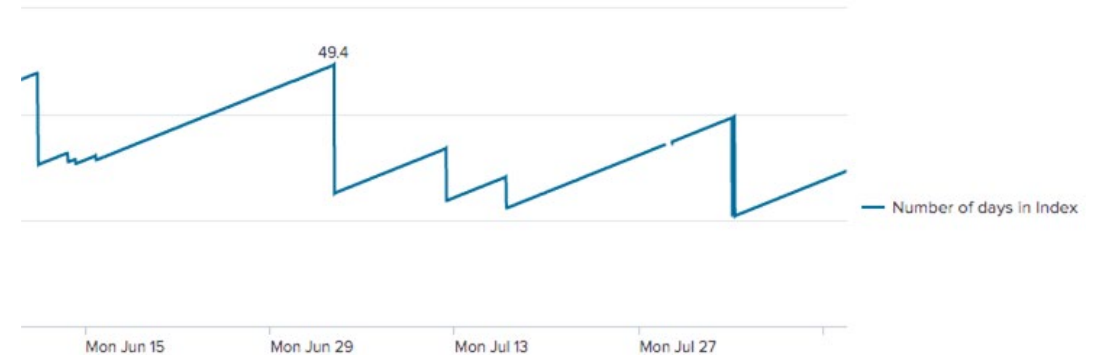
The alert condition for 'McciDataDeletionDetector' was triggered.

Check out the index's history using this report:
[redacted]
[redacted]
[redacted]

Alert: [McciDataDeletionDetector](#)
Trigger: Saved Search [McciDataDeletionDetector]: number of events (1)
[View results in Splunk](#)

Number of Days just removed from our index	Number of Days currently in our index
17.4	27.9

Visualization

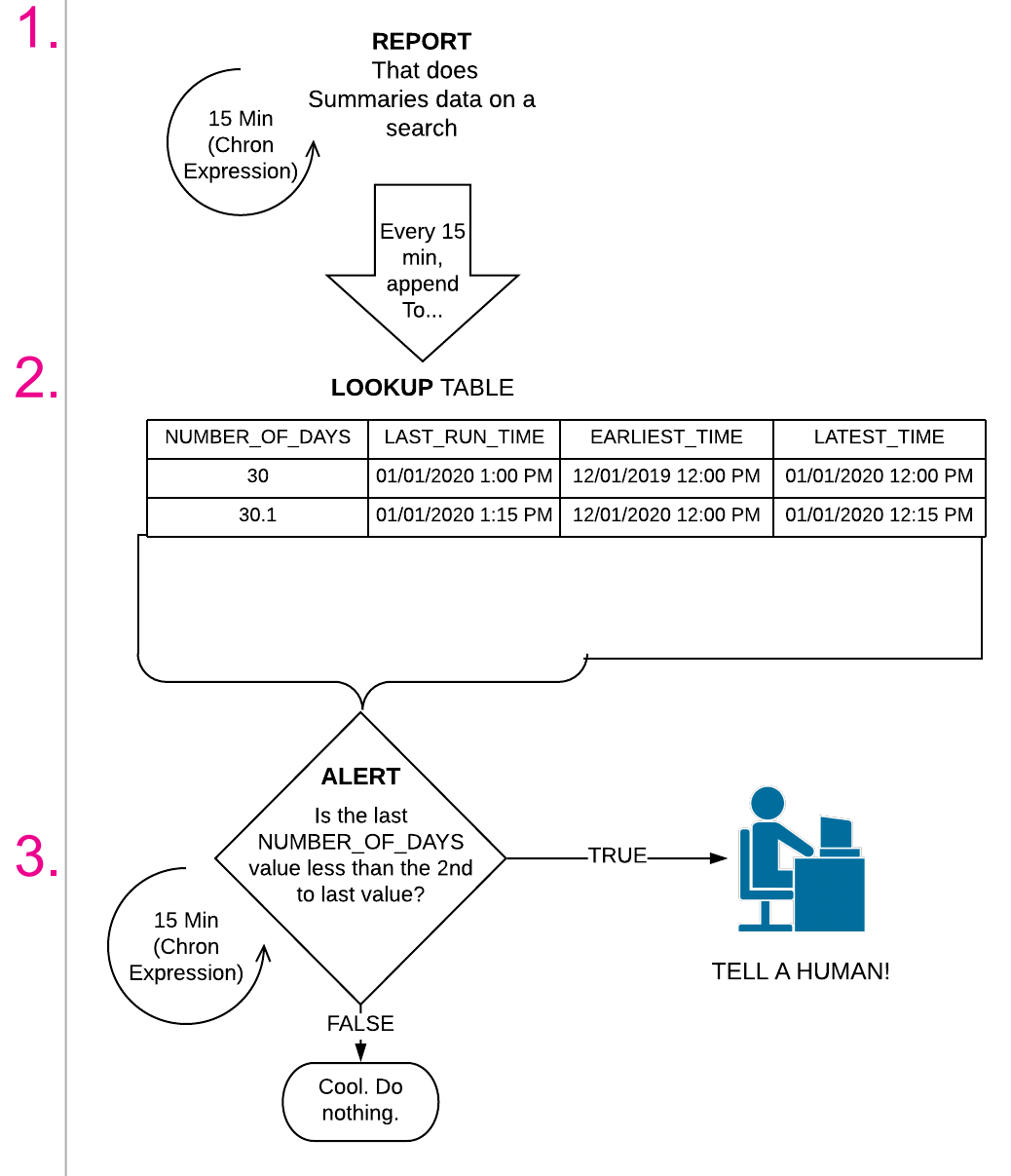


...armed with “the data”, I get to
bug MC Splunk team...

The Deletion Detector. Big Picture.

Three Components:

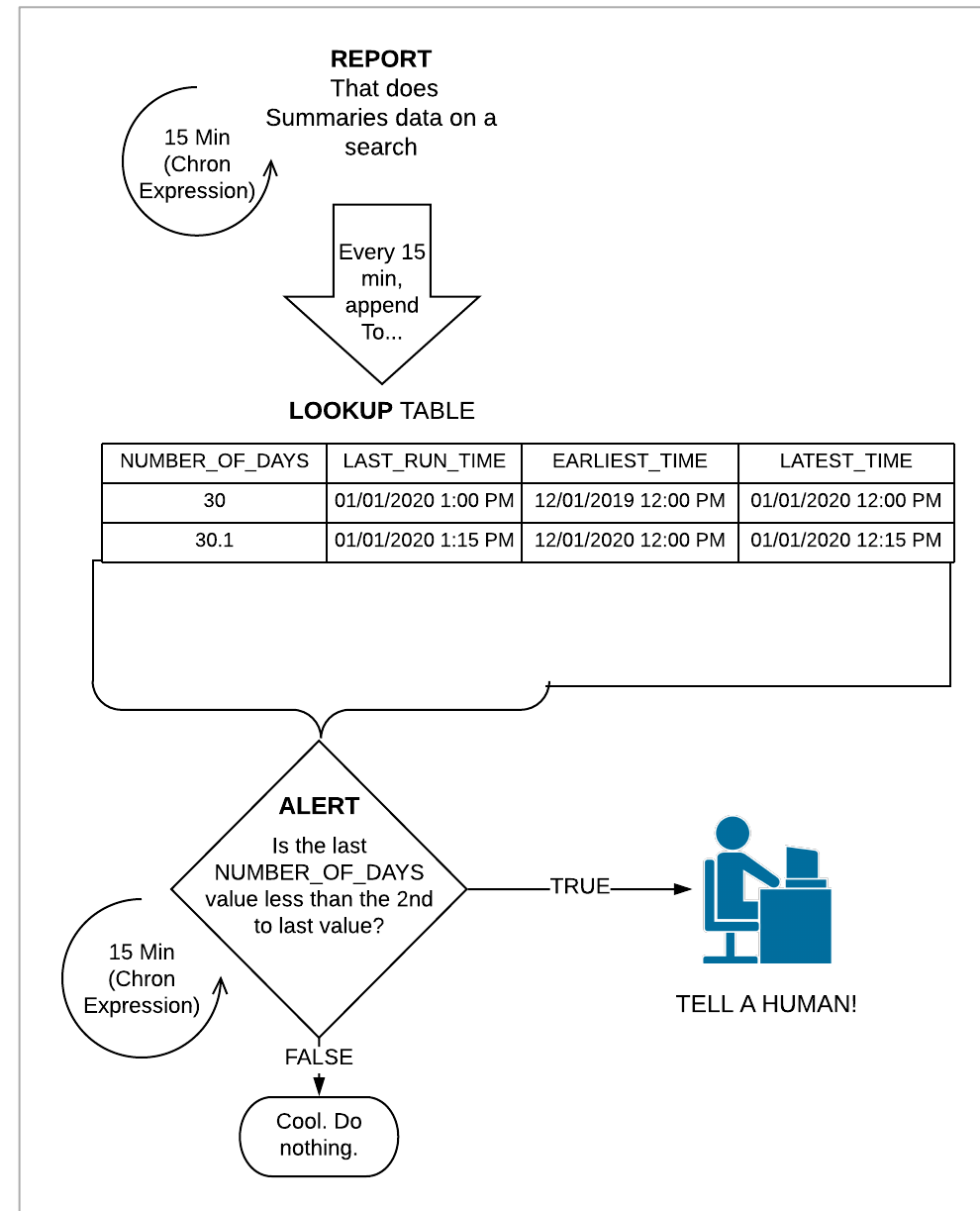
1. Report
2. Lookup Table
3. Alert



1. Report.

This report will only produce one row of data when run.

It summarizes how much data is in the index.



SPL for REPORT

```
index=BASE SEARCH  
| stats min(_time) as EarliestTime max(_time) as LatestTime  
count as NumberOfEvents  
...
```

SPL for REPORT

```
index=BASE SEARCH
| stats min(_time) as EarliestTime max(_time) as LatestTime
count as NumberOfEvents
| eval EarliestTime_Human = strftime(EarliestTime,"%Y-%m-%d
%H:%M:%S.%Q"), LatestTime_Human = strftime(LatestTime,"%Y-
%m-%d %H:%M:%S.%Q")
...
```


SPL for REPORT

```
index=BASE SEARCH
| stats min(_time) as EarliestTime max(_time) as LatestTime
count as NumberOfEvents
| eval EarliestTime_Human = strftime(EarliestTime,"%Y-%m-%d
%H:%M:%S.%Q"), LatestTime_Human = strftime(LatestTime,"%Y-
%m-%d %H:%M:%S.%Q")
| eval NumberOfDaysInIndex = round((LatestTime -
EarliestTime) / (60 * 60 * 24), 1)
...
```

SPL for REPORT

```
index=BASE SEARCH
| stats min(_time) as EarliestTime max(_time) as LatestTime
count as NumberOfEvents
| eval EarliestTime_Human = strftime(EarliestTime,"%Y-%m-%d
%H:%M:%S.%Q"), LatestTime_Human = strftime(LatestTime,"%Y-
%m-%d %H:%M:%S.%Q")
| eval NumberOfDaysInIndex = round((LatestTime -
EarliestTime) / (60 * 60 * 24), 1)
| eval LastRunTime = now(), LastRunTime_Human =
strftime(LastRunTime,"%Y-%m-%d %H:%M:%S.%Q")
...
```

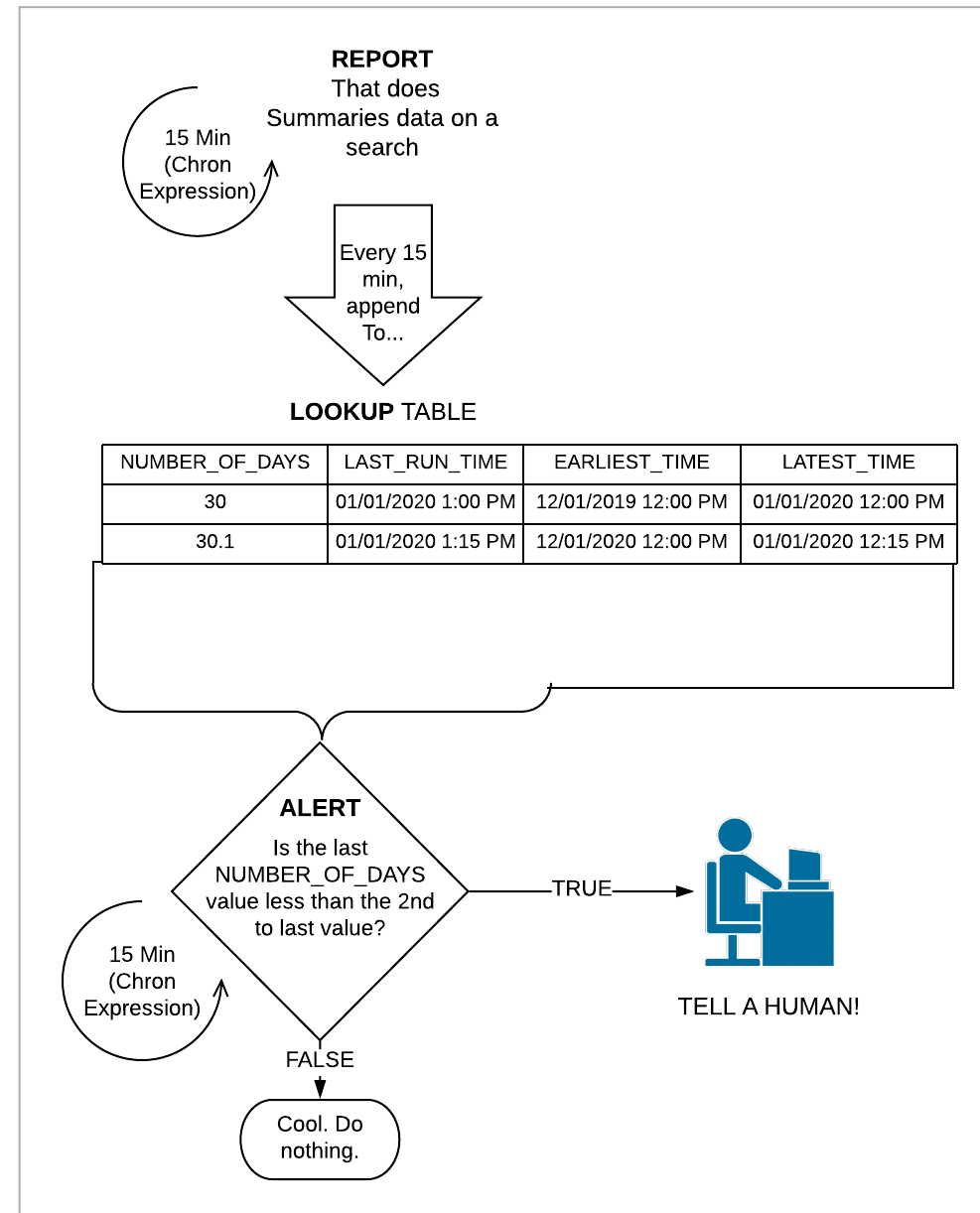
SPL for REPORT

```
index=BASE SEARCH
| stats min(_time) as EarliestTime max(_time) as LatestTime
count as NumberOfEvents
| eval EarliestTime_Human = strftime(EarliestTime,"%Y-%m-%d
%H:%M:%S.%Q"), LatestTime_Human = strftime(LatestTime,"%Y-
%m-%d %H:%M:%S.%Q")
| eval NumberOfDaysInIndex = round((LatestTime -
EarliestTime) / (60 * 60 * 24), 1)
| eval LastRunTime = now(), LastRunTime_Human =
strftime(LastRunTime,"%Y-%m-%d %H:%M:%S.%Q")
| table EarliestTime_Human LatestTime_Human
LastRunTime_Human NumberOfEvents NumberOfDaysInIndex
EarliestTime LatestTime LastRunTime
```


2. Lookup Table.

The Lookup table will store the results of the Report over time.

This is how we get that sweet, sweet context.



SPL for REPORT to LOOKUP table

```
| inputlookup indexHistory.csv  
| where LastRunTime > relative_time(now(), "-180d")  
| append  
|  
| ...  
|  
| outputlookup indexHistory.csv
```

SPL for REPORT to LOOKUP table

```
| inputlookup indexHistory.csv
| where LastRunTime > relative_time(now(), "-180d")
| append
  [ search index=BASE SEARCH
    | stats min(_time) as EarliestTime max(_time) as LatestTime
    | eval EarliestTime_Human = strftime(EarliestTime,"%Y-%m-%d
%H:%M:%S.%Q"), LatestTime_Human = strftime(LatestTime,"%Y-%m-%d
%H:%M:%S.%Q")
    | eval NumberOfDaysInIndex = round((LatestTime - EarliestTime) /
(60 * 60 * 24), 1)
    | eval LastRunTime = now(), LastRunTime_Human =
strftime(LastRunTime,"%Y-%m-%d %H:%M:%S.%Q")
    | table EarliestTime_Human LatestTime_Human LastRunTime_Human
NumberOfDaysInIndex EarliestTime LatestTime LastRunTime
  ]
| outputlookup indexHistory.csv
```


Lookup Table Results

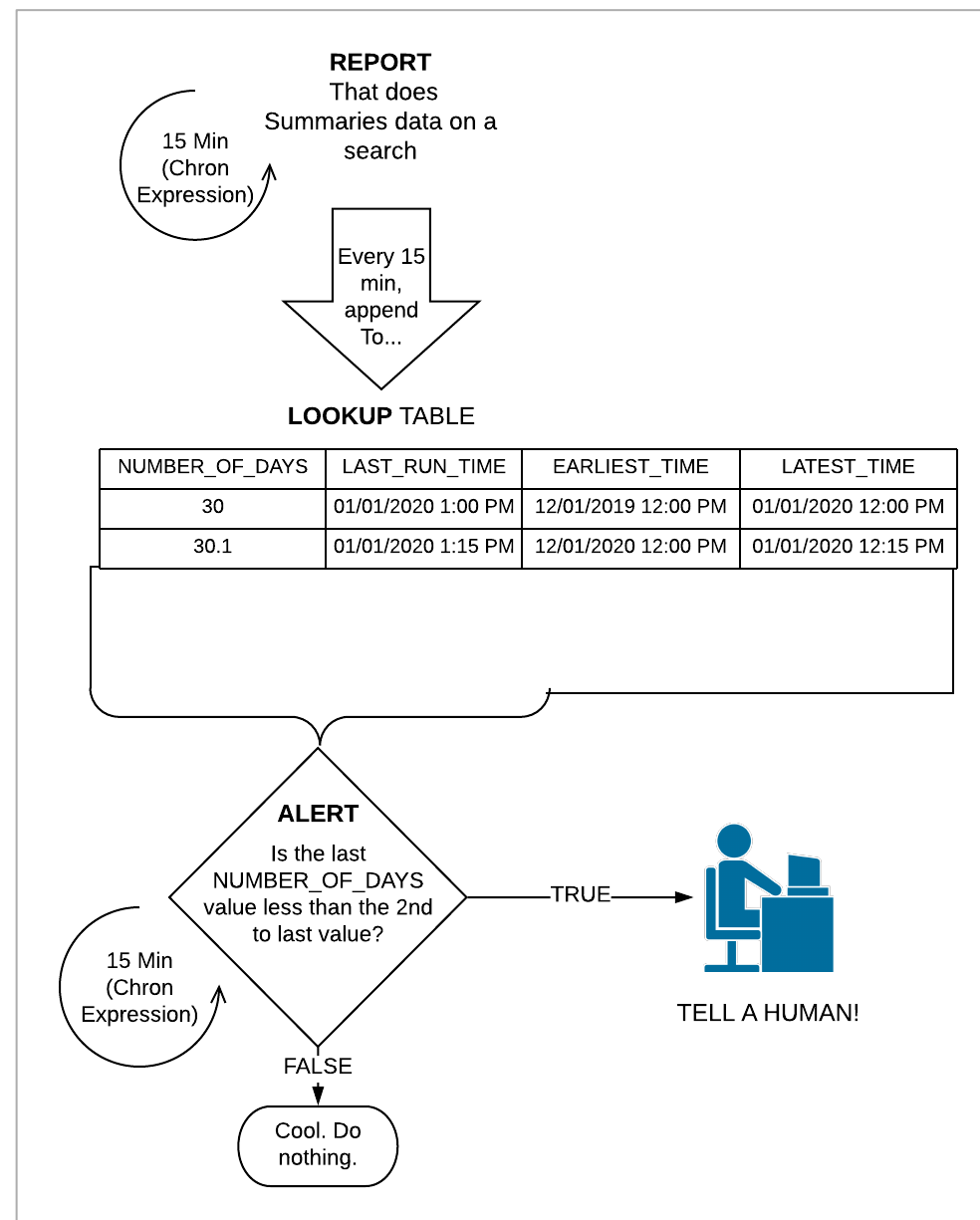
This will be the basis for the alert we create

EarliestTime ↕ /	EarliestTime_Human ↕ /	LastRunTime ↕ /	LastRunTime_Human ↕ /	LatestTime ↕ /	LatestTime_Human ↕ /	NumberOfDaysInIndex ↕ /	NumberOfEvents ↕ /
1594648276	2020-07-13 09:51:16.000	1596432600	2020-08-03 01:30:00.000	1596432599	2020-08-03 01:29:59.000	20.7	41066
1594648276	2020-07-13 09:51:16.000	1596448800	2020-08-03 06:00:00.000	1596448657	2020-08-03 05:57:37.000	20.8	61306
1594648276	2020-07-13 09:51:16.000	1596449700	2020-08-03 06:15:00.000	1596449652	2020-08-03 06:14:12.000	20.8	61327
1594648276	2020-07-13 09:51:16.000	1596450600	2020-08-03 06:30:00.000	1596450599	2020-08-03 06:29:59.000	20.9	61392
1594648276	2020-07-13 09:51:16.000	1596451500	2020-08-03 06:45:00.000	1596451238	2020-08-03 06:40:38.000	20.9	61399
1594648276	2020-07-13 09:51:16.000	1596452400	2020-08-03 07:00:00.000	1596452257	2020-08-03 06:57:37.000	20.9	61405
1594648276	2020-07-13 09:51:16.000	1596453300	2020-08-03 07:15:00.000	1596453247	2020-08-03 07:14:07.000	20.9	61438

3. Alert.

The Alert is how we get notified.

Based off the data in the lookup table, did a deletion just occur?



SPL for Alert

```
| inputlookup indexHistory.csv  
| sort 3 -LastRunTime  
| sort 0 LastRunTime  
...
```

SPL for Alert

```
| inputlookup indexHistory.csv  
| sort 3 -LastRunTime  
| sort 0 LastRunTime  
| streamstats current=f last(NumberOfDaysInIndex) as  
prevDayCount  
...
```

SPL for Alert

```
| inputlookup indexHistory.csv
| sort 3 -LastRunTime
| sort 0 LastRunTime
| streamstats current=f last(NumberOfDaysInIndex) as
prevDayCount
| where prevDayCount > NumberOfDaysInIndex AND
NumberOfDaysInIndex < 180
...
```


SPL for Alert

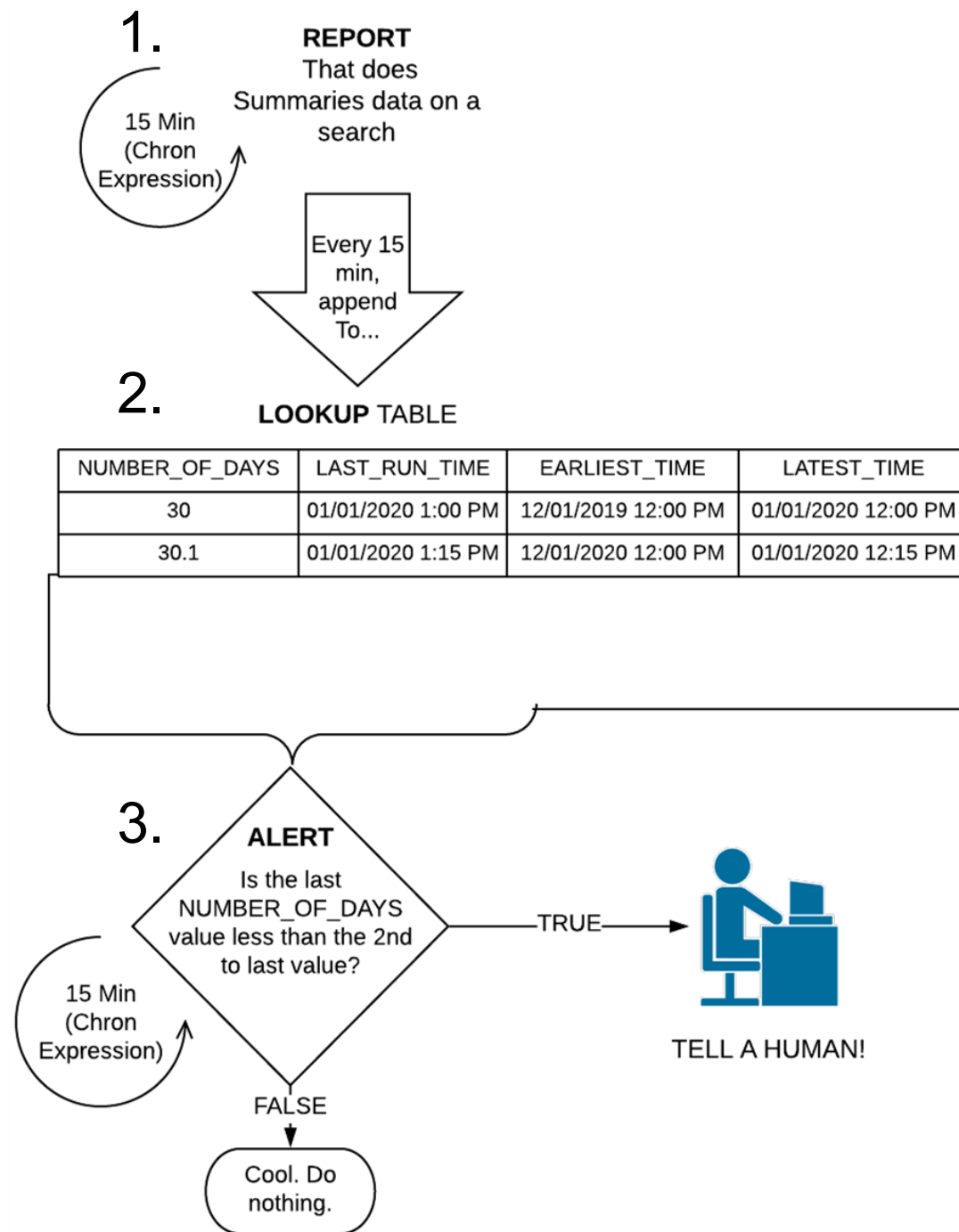
```
| inputlookup indexHistory.csv
| sort 3 -LastRunTime
| sort 0 LastRunTime
| streamstats current=f last(NumberOfDaysInIndex) as
prevDayCount
| where prevDayCount > NumberOfDaysInIndex AND
NumberOfDaysInIndex < 180
| eval DaysRemoved = prevDayCount - NumberOfDaysInIndex
...
```

SPL for Alert

```
| inputlookup indexHistory.csv
| sort 3 -LastRunTime
| sort 0 LastRunTime
| streamstats current=f last(NumberOfDaysInIndex) as
prevDayCount
| where prevDayCount > NumberOfDaysInIndex AND
NumberOfDaysInIndex < 180
| eval DaysRemoved = prevDayCount - NumberOfDaysInIndex
| rename DaysRemoved AS "Number of Days just removed from our
index", NumberOfDaysInIndex AS "Number of Days currently in
our index"
| table "Number of Days just removed from our index" "Number
of Days currently in our index"
```

The Deletion Detector:

Putting it all together



Slack Notification

Data loss is sad! :(

The screenshot shows a Slack message from a channel named 'ci-notify'. The message is an email notification from the 'email' app, received at 6:20 PM on Monday, June 8th. The email subject is 'Splunk Alert: McciDataDeletionDetector' and the date is 'Jun 8th'. The body of the email states: 'The alert condition for 'McciDataDeletionDetector' was triggered. Check out the index's history using this report: [redacted]'. Below this, it says 'Alert: McciDataDeletionDetector' and 'Trigger: Saved Search [McciDataDeletionDetector]: number of events (1)'. There is a link 'View results in Splunk'. A table follows with two columns: 'Number of Days just removed from our index' and 'Number of Days currently in our index'. The first column has a value of 17.4 and the second has 27.9. The email concludes with 'If you believe you've received this email in error, please see your Splunk administrator.' and a footer line 'splunk > the engine for machine data'. At the bottom of the Slack message, there are reaction buttons showing one '😞' (sad face) and one '😬' (grimacing face) reaction.

Number of Days just removed from our index	Number of Days currently in our index
17.4	27.9

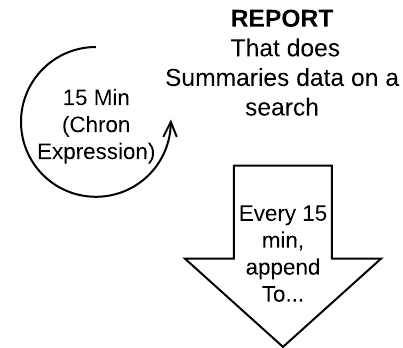
Turns Out Our Splunk Infrastructure Couldn't Handle the Data Load.

Moving to a new environment we got our retention back up to 180 days.

The Deletion Detector... or Is It More Than That?

Alerts that
require
context!

1.

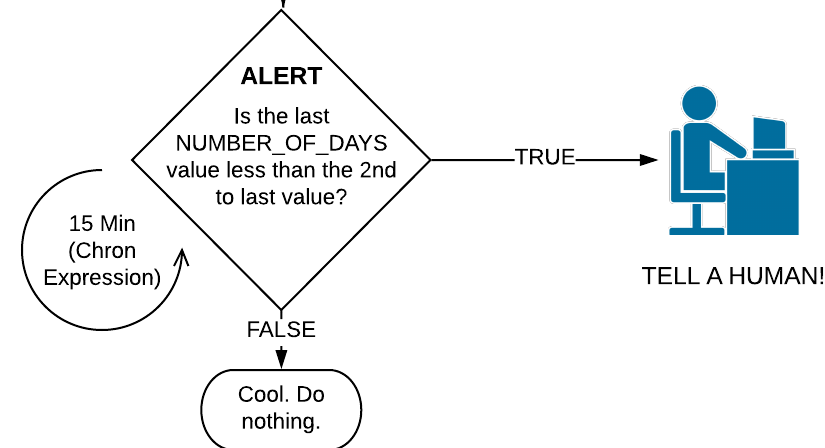


2.

LOOKUP TABLE

NUMBER_OF_DAYS	LAST_RUN_TIME	EARLIEST_TIME	LATEST_TIME
30	01/01/2020 1:00 PM	12/01/2019 12:00 PM	01/01/2020 12:00 PM
30.1	01/01/2020 1:15 PM	12/01/2020 12:00 PM	01/01/2020 12:15 PM

3.



Lessons Learned

- When I have an alert that requires **context**, I have a formula for solving the problem
- It's good solution for short term problems but consider when to use summary indexes

