Day - 1

13 January 2025 11:25

1. Splunk Introduction
2. Component of splunk.
3. Use Cases splunk.
4. UI of splunic.
5. Add the data in splunk.

---

1. Introduction:-

(1) Monitoring → Logs in splunk config App.

1. Dashboards
2. Alert
3. Report
4. Prediction.
5. Knowledge object
6. Splunk MLTK (Machine Learning Toolkit)

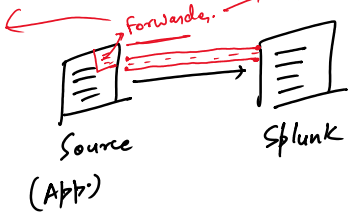
→ User level login
→ Alert

App

1. All the sources
2. All the type of data.
   1. Structured → csv, json
   2. Semi-structured → xml
   3. Unstructured data. → Mix, log, txt

Component:-

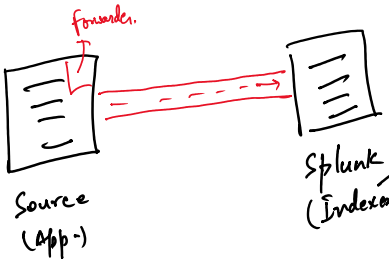
1. Forwarder →
2. Indexer →
3. Search Head. → GUI where end user goes search the data, make the dashboard, Alert & report.
4. License Master →



pipeline that will forward the logs to the splunk.

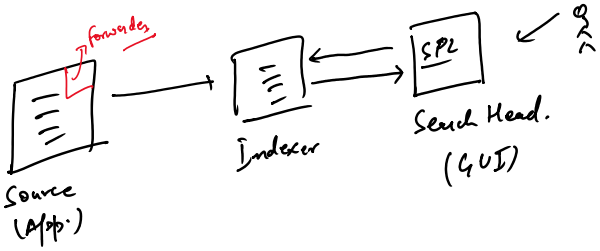
(1) What!
(2) Where?
(3) Basic config. Stanza.

Indexer:-



Internal Database where the incoming logs from the source will be stored over there.

Search Head!-



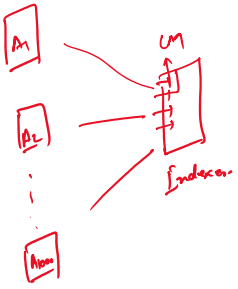
License Master!- What is the parameter splunk is going to change!

Agent that will make sure. that you will consume or Not going license breach.

1. Amount of Data. per Day wise
2. No. of user.
3. No. of Sources.
4.

5GB/day → 1 year.

$$$



Limitation:-

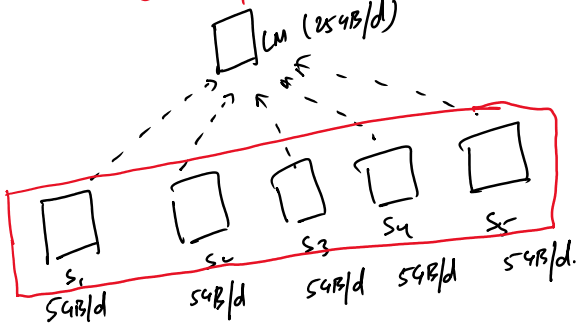
① 30days - 5 breach.

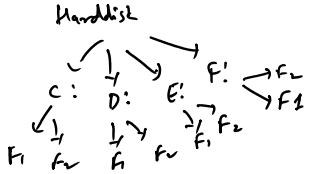
① Up front payment.
② Data is safe.
③ Search, will be disabled

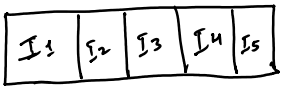
License Tooling:-

$S_1+S_2+S_3+S_4+S_5$ *25GB

① License less cost.
② Limition/ upperlimit issue will reload.
③ Management of license.







Indexer

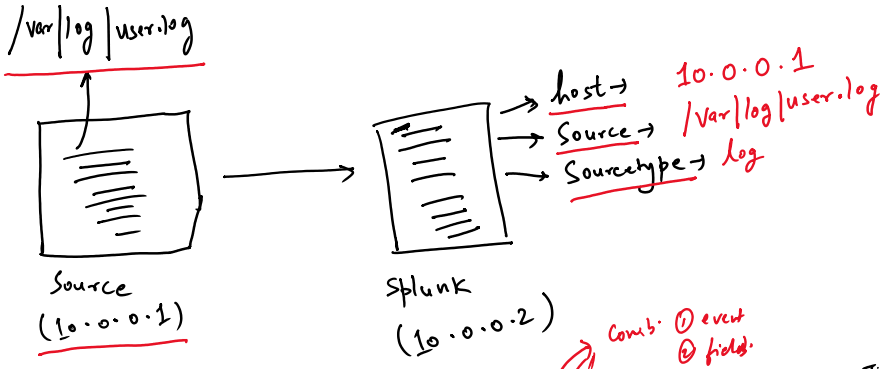
3 Types of Index:- Reserved to store splunk App.log.

① Predefined Index:- * (_internal, _audit, _telemetry, #)
② Default Index → main (License is consumed, Default data goes over there)
③ Custom Index → VK-snow, sample_idx (Own App. specific data)

No License is consumed.

License Consumed.

License is consumed.



Smart — Contd: ① event ② fields.

① fetch the event
② Extract the field.

Fast (Min. Time)

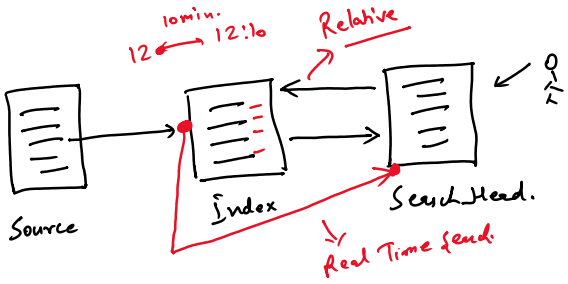
① fetching the events.

index= _internal

1. fetch all the events
2. Extract the fields.

Verbose:- (Max. time)

① All the details of the events & all the fields.

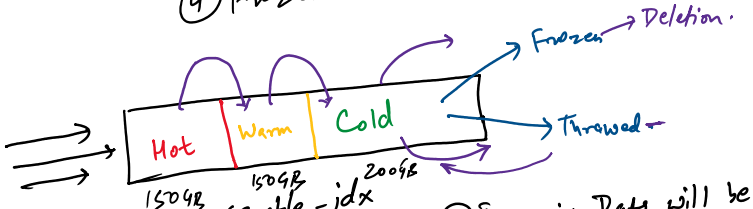


Relative → Historical Data
Real Time →

⑤ Thrawed Bucket.

Bucket:-

① Hot Bucket
② Warm Bucket
③ Cold Bucket
④ Frozen Bucket



① Incoming Data will be coming in Hot Bucket.

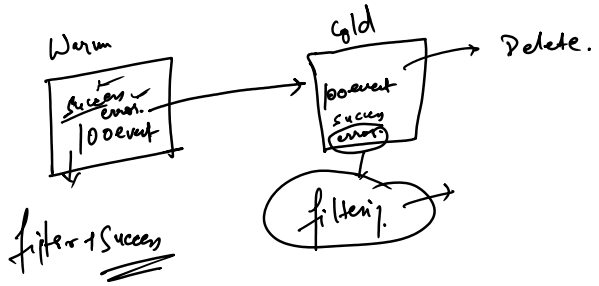
② Movement of Data across Multiple Bucket

Anyone true, Data will move. ⟵ { ⓐ Size of Bucket. ⓑ Age of Data.

Hot - 150GB, 7d
↓ 100GB
Warm

Naming Convention for index name:-

① No digit
② No space
③ All lowercase string.



filter + Success

Basic SPL Query:-

1. Table
2. Rename.
3. Stats
4. Eval
5. top
6. Rare.
7. Sort
8. Dedup
9. Fillnull

(1) Table:- Tabular output.

Syntax:- |table field1 field2 field3

fieldname – Case Sensitive
fieldvalue – Case Insensitive.

(2.) Rename:- Create the new name of the fields.

Syntax:- |rename old_name AS new_name.
ticket_number → Incident_name.

(3.) Stats:- Statistical output.

(1) Count → Number of event.
(2) Sum → Summation of the Numeric Value. → |stats sum(—)
(3) Avg → Avg. of the Numeric Value.
(4) List → Category the data on the basis of all field.
(5) Values → " " " " Unique field.

Fillnull → Fix Blank spaces.

|fillnull
By default, it will add 0 at the blank space.

|fillnull value="NULL"

a	b	c
—	—	—
—	—	—

Sor

Sort:- Ascending → Sort field or sort +field
Desending → Sort - field.

Eval:- Do the evaluation Activity.

int a, var a, str b

eval a = ____

1. Calculation.
2. if-else. Condition.
3. Case Statement

1. Calculation → Bytes ⟶ Kb

Kb = (bytes/1024)

eval Kb = (bytes/1024)

2. **if-else:-**

```
if (a>b)
    {
    Print(a);
    }
else {
    Print(b);
    }
```

→ if (a>b, "a", "b")

if (Condition, True, False)

3. **Case Statement:-**

Switch(0): Sunday
Switch(1): —
(2): —
(3): —

Case (Condition 1, "V1", Condition 2, "V2", Condition", V3, . . .
. . . , 1=1, "____")
↑
Universal

(3): —

(4): —

default (1=1): —

↳ True.

↑
Universal
Condition

Sev = 1, 2, 3, 4

1 → Critical
2 → Normal
3 → Low
4 → Info.

(5.) **Top:-** Top Command give the max. top value

|top Sourcetype

By default → return top 10 Value

|top limit = 3 Sourcetype

top 3 Sourcetype.

|top limit = 0 Sourcetype

Unlimited Value.

Rare:- Top minimum / least Value.

|rare Sourcetype → least sourcetype.

|rare sourcetype limit=3

↳ least 3 sourcetype.

|rare limit=0 Sourcetype

↳ least all sourcetype.

top/rare → Sourcetype, count, percent.

top/rare → Sourcetype, count, percent.

fields → What field to be included or excluded from the output.

| fields - percent → exclude the field named percent from the output.

Visualization:- Chart Command.

asc def

as..f

| chart count by current_ticket_state

count → y-axis

current_ticket_state → x-axis.