Day - 1

02 December 2024 10:06

1. Last 2000 → Data → Storage → Analysis

Storage:
1. Data Warehouse
2. Data lake.

Data Swamp

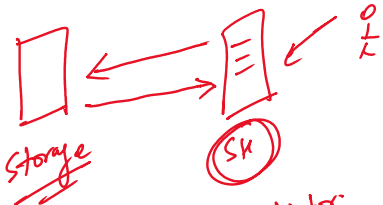
Analysis:
1. ELK
2. splunk
3. DataDog.
4. Nagios.
5. Dynatrace.

1. Splunk:- Data Analysis, Monitoring, Dashboard, Report, Alert, Prediction, MLTK Toolkit
   1. Any Sources.
   2. Any Data Type.

Infra.
1. CPU
2. Memory
3. Disk.
4. Process

Adv:-
1. Real Time Monitoring.
2. Prediction.
3. Visualization.
4. Alert & Report.
5. Any Source & Any Type of Data
6. Learning.
7. Customer Support.

Dis:-
1. Expensive license.
2. Hardware Pre-requisite.



1. Size/Amount of Data.
2. Query Writing [Append, join]

splunk phantom

IT Service Intelligence:
1. splunk ES.
2. splunk SOAR
3. splunk ITSI
4. splunk APM

5. splunk Victor.
6. splunk observability
7. splunk MLTK.
8. splunk VBA

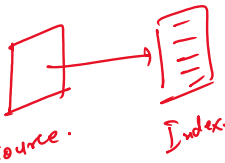
* Components of splunk:-

1. Indexer
2. Forwarder.
3. Search Head.
4. License Master.
5. Deployment Server.
6. cluster Master.
7. Deployer

→ Management Instances.

(1) Indexer:- Storage unit where you are going to store incoming data.

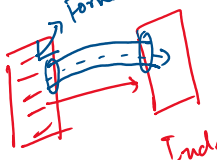
1. Parsing } → optional
2. storage



Indexer

(2) Forwarders:-

Forwarder → Package help to create the pipeline, that will help send the data from source to destination.



Indexer.

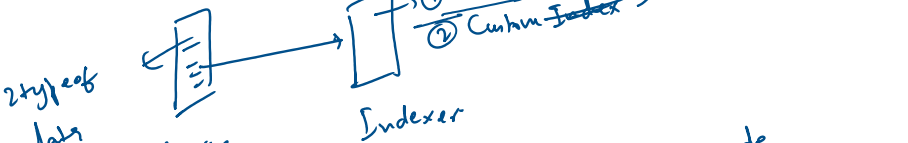
splunkd.log → log file.

1. ping
2. telnet ip 9997
3. Access of file.

/var/log/splunk/splunkd.log.

GUI

index=_internal source=



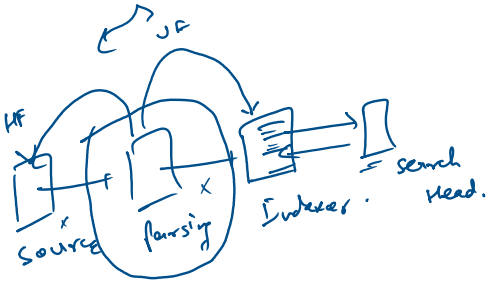
(1.2GB Tar, UNIX-5.8GB)

splunk Enterprise Package.

1. Heavy forwarder → Parsing option.
2. Universal forwarder → No Parsing.
3. Light forwarder

Obsolete

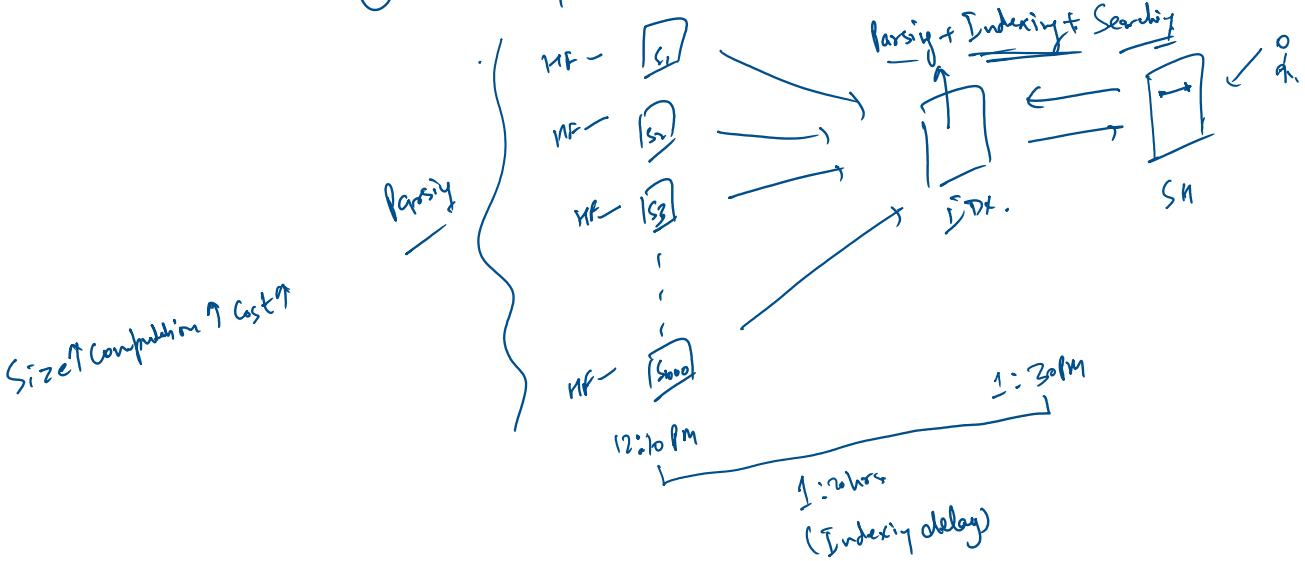
Splunk UF. [35MB, Tar 150MB, Unix]



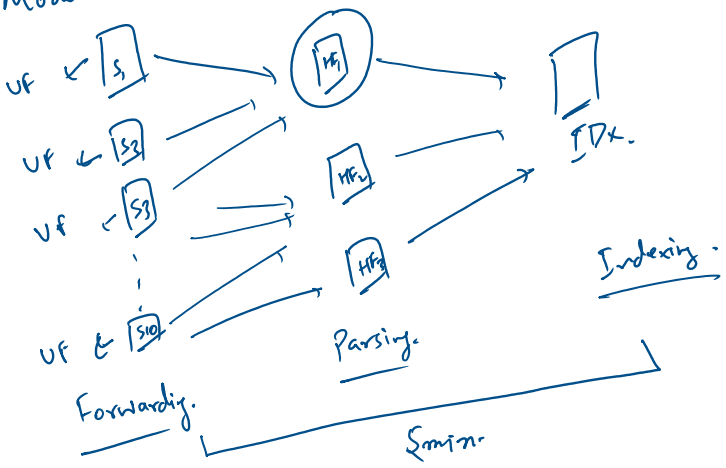
1. Heavy Forwarder:-

Adv.:-
1. Parsing.
2. Less Network Traffic.
3. Masking can be done at source end.
4. Less Load at Indexer.

Disadv.:-
1. Resource Consumption will High
2. Cost will increase
3. Forwarding from source will be less speed as Parsing will take sometime to complete.



1. Load ↑ HF, Load ↓ UF
2. Criticality of Data. use HF to save indexing time.
3. Hybrid Model.



... Search Query, Create Dashboards, Report &

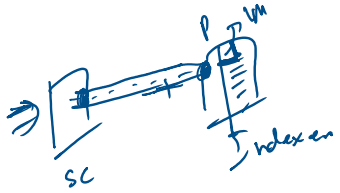
5min.

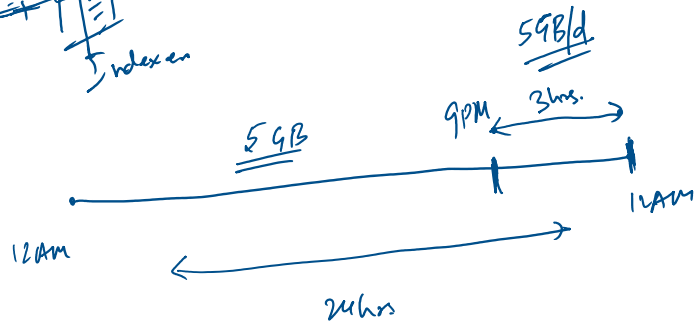
(3) Search Head:- GUI where end-user will write the search Query, Create Dashboards, Report & Alerts.

(4) License Master:-

(1) No. of user. ∝
(2) No. of machine. ∝
(3) No. of sources ∝
(4) Amount of Data → Ingest in a single day

5GB/day → 1 year. → License Master → Agent make sure to consume data within a limit only.

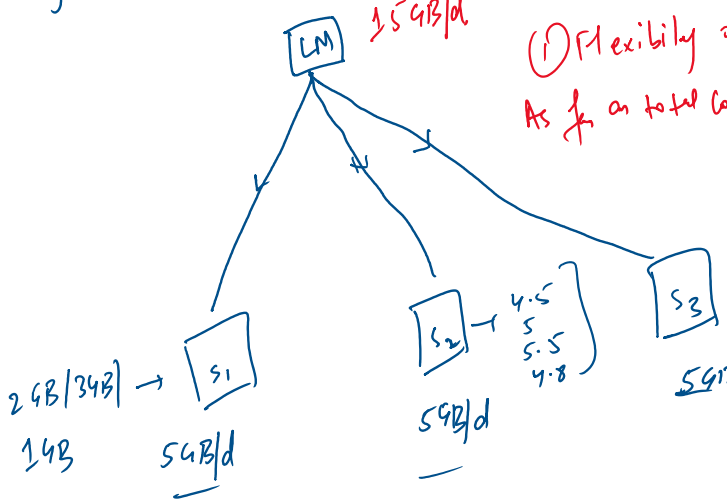




(1) Data will come.
(2) Indexing will happen. → Safe Data.
(3) Searching will be stopped → Stale Mode → Dashboard, Report, Alert, Knowledge object → No Searching + None of the feature will work.

(1) 5 time → 30 day Windows

License Pooling:-



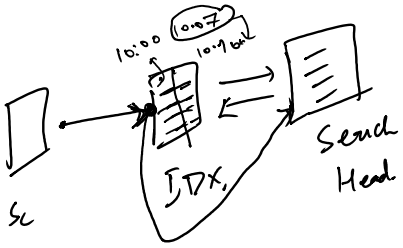
(1) Flexibility of the data usage server wise.
As for as total consumption ≯ 15GB
S1 + S2 + S3 ≯ 15GB
S1 → 2GB
S2 → 8GB
S3 → 5GB
S1 → 1GB
S2 → 7GB
S3 → 7GB

(2) Buy license in Bulk, Per GB cost will be less

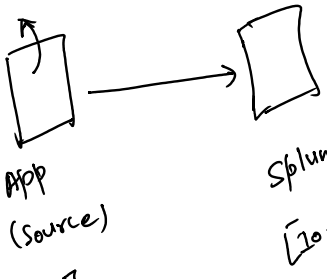
Indexer → Multi Index

Index
- Pre-defined Index → _internal, _audit, _introspection, _thefishbucket, Splunk Application logs → No License charged. (→ No license charged.)
- Default Index → main → License is consumed
- Custom Index → VK_snow, source_access, aws_logger. License consumed.

License Consumed

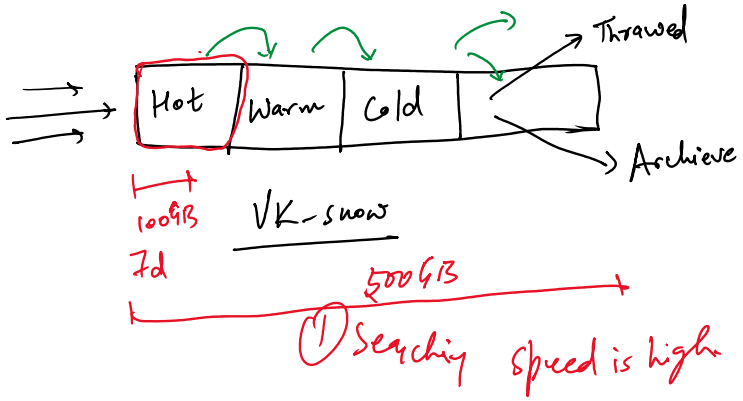


Log Path: /var/tmp/abc.log



Source → /var/tmp/abc.log.
Host → 10.0.0.1
sourcetype → log → Data type.
time → log_generated_time, index time.

Bucket in Index:-



1. Size of Bucket
2. Age of Data.

(either will be true.)

① Searching speed is high → last 24hrs.

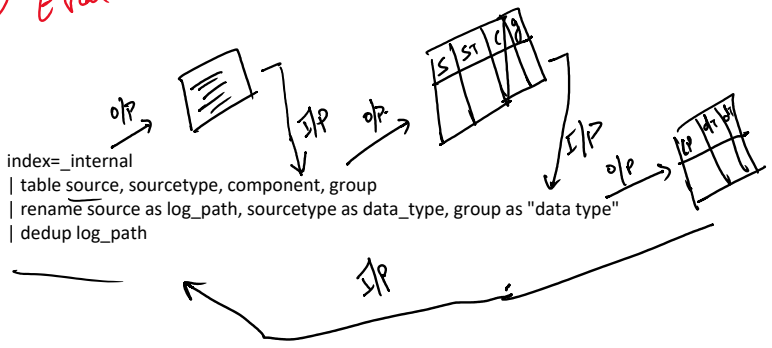
Commands (SPL):-

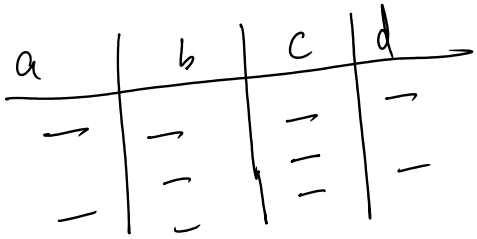
① table → Tabular output Syn:- | table fields, field2, field3

② rename → change the name of field. search time change. Syn:- | rename old_name AS new_name.

1. table → Tabular output
2. Rename → change the name of field. search time change. Syn:- |rename
3. dedup → Remove the Duplicate values. syn:- |dedup field_name
4. stats → count, dc, sum, avg, list, values → statistical output.
5. fillnull → Handle the blank places |fillnull value = NA fields
6. append | appendcol | appendpipe.
7. join.
8. addcoltotal | addtotal.
9. Rex
10. chart
11. timechart
12. eventcount.
13. Top | Rare.
14. where | search.
15. field.
16. Eval → Calculation, if-else, Case ⇢ Evaluation Command. Initialize the variable.

int, str, var → Programming language





| fillnull value = NA d

7

True.

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

→ if (a>b, a, b)

↑ True.

↳ false.

↓

Conditional Statement

```
Switch (a);
    (1) :
    (c) :
default :
```

Case (Condition 1, Value1, Cond 2, Value 2, Cond 3, Value 3, default, Valued)