

# Splunk4Admins - Indexer Performance

## Lab Guide

### Overview

This lab guide contains the hands-on exercises for the Splunk4Admins - Indexer Performance workshop. Before proceeding with these exercises, please ensure that you have a copy of the workshop slide deck, which will help to put into context the tasks you are carrying out.

Download the workshop slide deck: <https://splk.it/S4A-IDX-Attendee>

### Prerequisites

In order to complete these exercises, you will need your own Splunk instance. Splunk's hands-on workshops are delivered via the [Splunk Show portal](#) and you will need a splunk.com account in order to access this.

If you don't already have a Splunk.com account, please create one [here](#) before proceeding with the rest of the workshop.

### Troubleshooting Connectivity

If you experience connectivity issues with accessing either your workshop environment or the event page, please try the following troubleshooting steps. If you still experience issues please reach out to the team running your workshop.

- **Use Google Chrome** (if you're not already)
- If the event page (i.e. <https://show.splunk.com/event/<eventID>>) didn't load when you clicked on the link, try **refreshing the page**
- **Disconnect from VPN** (if you're using one)
- **Clear your browser cache and restart your browser** (if using Google Chrome, go to: Settings > Privacy and security > Clear browsing data)
- **Try using private browsing mode** (e.g. Incognito in Google Chrome) to rule out any cache issues
- **Try using another computer** such as your personal computer - all you need is a web browser! Cloud platforms like AWS can often be blocked on corporate laptops.

## Table of Contents

<b>Overview.....</b>	<b>1</b>
<b>A Note About Outputs.....</b>	<b>4</b>
<b>Exercise 1 – Explore the MC Overview.....</b>	<b>4</b>
Description.....	4
Steps.....	4
Access the Workshop MC GUI.....	4
Login to the GUI.....	5
Access the “jumpshot” and then the Splunk systems command line interface.....	5
On the MC GUI Observe the Alert Dropdown.....	6
Access the MC Overview Dashboard.....	7
<b>Exercise 2 – Explore the Indexing Performance Dashboards.....</b>	<b>9</b>
Description.....	9
Steps.....	9
On the MC GUI.....	9
Instances by Indexing Rate panel.....	10
Deployment-Wide Indexing Rate panel.....	11
Instance by Median Queue Fill Ratio panel.....	11
Deployment-Wide Median Queue Fill Ratio panel.....	12
<b>Exercise 3 – Explore diagnosing slow indexing.....</b>	<b>13</b>
Description.....	13
Steps.....	13
Use the already running GUI on the MC, access the IP:Deployment Dashboard.....	13
Go to the Indexer Performance:Advanced dashboard.....	14
Go to the Resource Usage:Machine dashboard.....	14
Explore the issue from the OS.....	15
Reset the nVM.....	16
<b>Exercise 4 – Explore a node reporting full memory.....</b>	<b>17</b>
Description.....	17
Steps.....	17
Use the already running GUI on the MC, access the Overview Dashboard.....	17
Go to the IP:Advanced dashboard.....	18
Go to the Resource Usage:Deployment dashboard.....	18
Go to the Resource Usage:Machine dashboard.....	19
Explore the issue from the OS.....	20
<b>Exercise 5 – Explore a node that is undersized for the workload.....</b>	<b>21</b>
Description.....	21
Steps.....	21
Use the already running GUI on the MC, access the Overview dashboard.....	21
Explore the issue from the OS.....	23
Remediation.....	24
<b>Exercise 6 – Explore network tools.....</b>	<b>25</b>

Description.....	25
Steps.....	25
Access the indexer 01 command line.....	25
Engage iftop.....	25
Access the Cluster Manager command line.....	26
Engage mtr.....	26
<b>Links.....</b>	<b>27</b>

## A Note About Outputs

This environment, much like any real production environment, is in flux. The various issues and triggers will cause differing values in the various dashboard panels and OS command line outputs. Don't be concerned if the values that are seen on the individual lab systems are different from those in either the presentation or the lab guide.

## Exercise 1 – Explore the MC Overview

### Description

Access the Monitoring Console (MC) GUI and explore the high-level Overview dashboard. Understand how the dashboard can lead to the next steps in diagnosing whatever performance issues are reported/experienced.

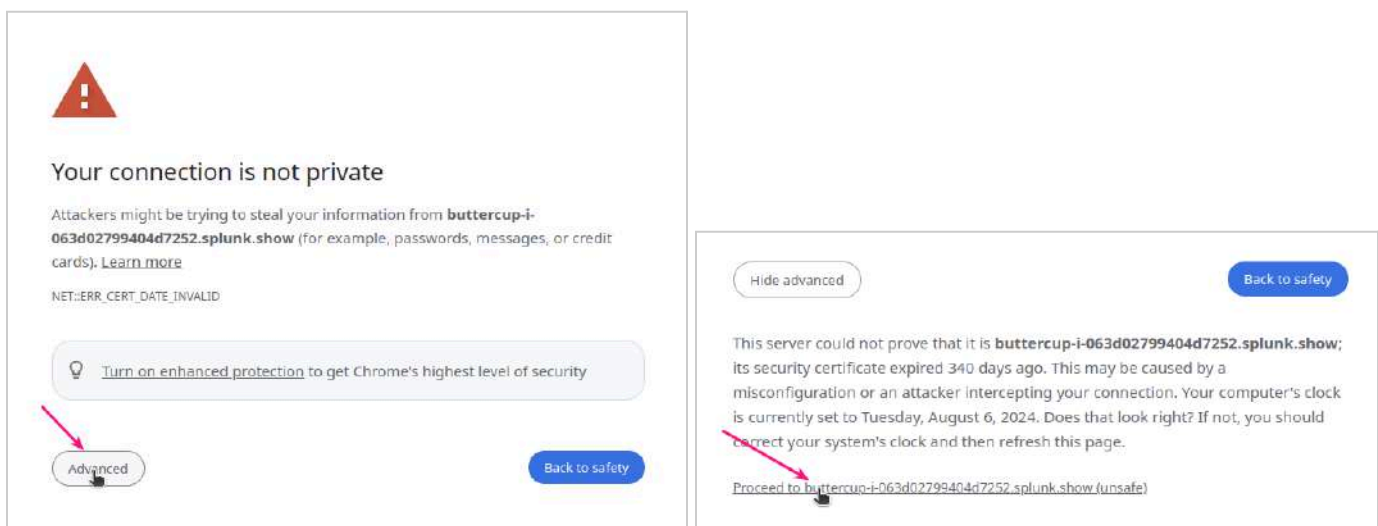
### Steps

#### Access the Workshop MC GUI

1. Access the environment by clicking on the Splunk Enterprise link provided in the Splunk Show interface or in the address bar of a browser window and typing “<your\_assigned\_IP>:4505”. The port is important, as it is different from the standard Splunk instance.

**Note:** Due to constraints within the lab environment, the connections will be unencrypted and unvalidated. This is not an issue, as nothing of value is passed, but you will see the following screen.

Click on the “Advanced” button and then on “Proceed to...” at the bottom of the prompt.



## Login to the GUI

2. Login using the credentials provided in Splunk Show.

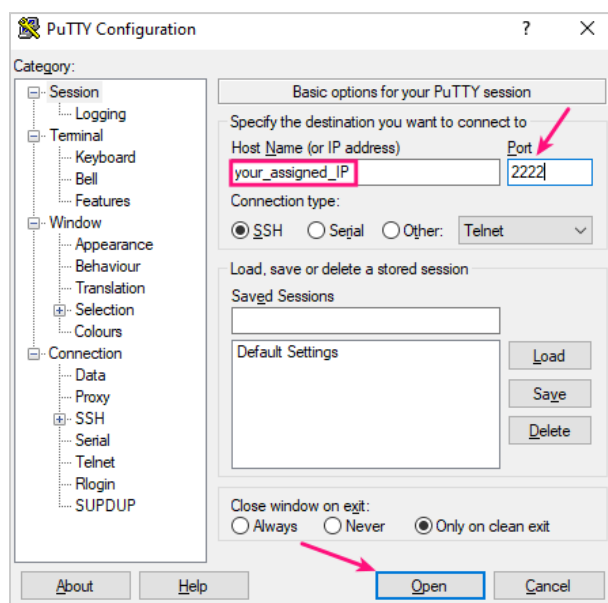


## Access the “jumphost” and then the Splunk systems command line interface




3. The command line version for both linux and macOS [note the lowercase “p” to change the port 2222] and change “<your\_assigned\_IP>” to be the actual assigned IP or hostname, with the "SSH Password" provided by Splunk Show.

```
ssh -p 2222 attendee@<your_assigned_IP>
```

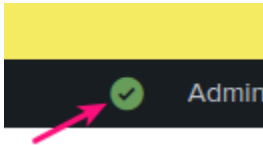
4. In Windows, here with PuTTY - note the GUI (change “<your\_assigned\_IP>” to be the actual assigned IP): same "SSH Password" as above from Splunk Show. Be sure to enter the login name as ‘attendee’ and change the port to ‘2222’.



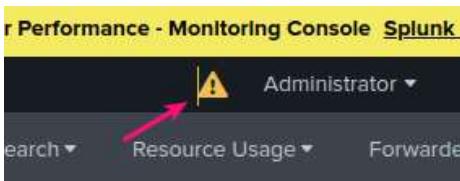
## On the MC GUI Observe the Alert Dropdown

5. At the top of the window, see a green circle check mark , a yellow triangle exclamation mark , or a red square  exclamation mark

6. Green means “healthy”



7. While yellow or red implies there are issues to be explored



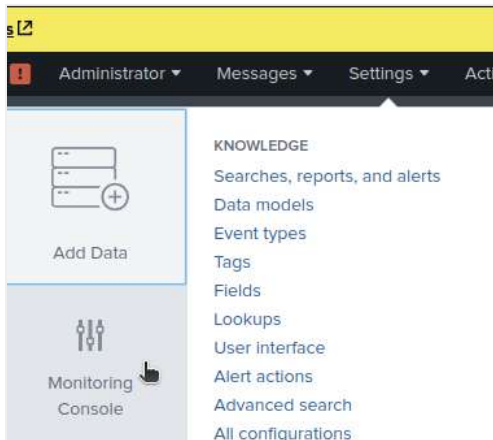
8. Clicking on the icon brings up a very high-level overview of detected issues



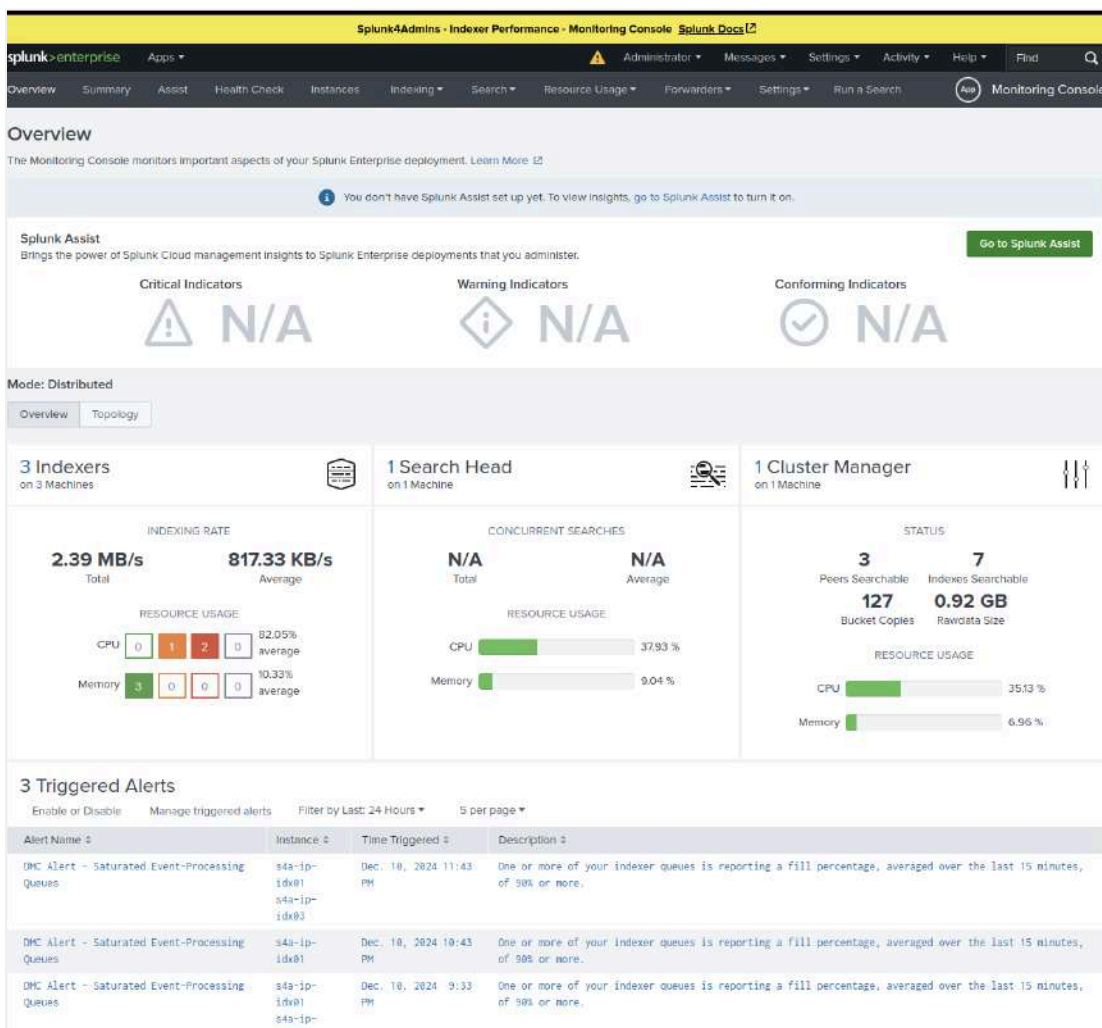
9. Clicking on each of these will cause the right side to provide additional details, including the triggers for the alert

## Access the MC Overview Dashboard

10. Click on the black bar “Settings”, then select “Monitoring Console”



11. This will bring you to the Overview dashboard



12. On the overview dashboard, look for the “red” indicators. They will begin to give an idea of what is occurring in the environment. Here we see the indexers are running “hot” in that they are showing “orange” and “red” for different
  13. At the bottom, in the table, alerts are available that help detect where issues may lie. These are configurable, but due to time constraints, this Workshop does not address configuring new alerts or tuning existing alerts on the MC.
  14. Click on any alerts and explore the output. The lab has a few things going wrong, so various issues will pop-up here. There are going to be different alerts fired at different times due to different detections.
-



## Exercise 2 – Explore the Indexing Performance Dashboards

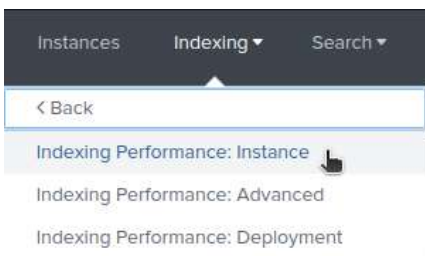
### Description

Experience the ways that the MC is able to help diagnose indexing performance issues.

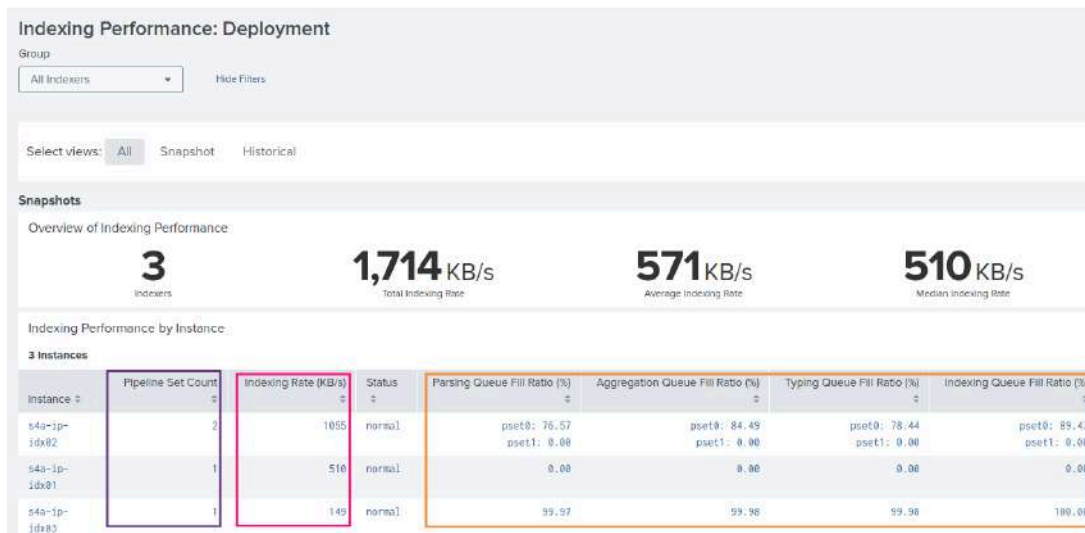
### Steps

#### On the MC GUI

1. Select on the gray-bar “Indexing”, then “Performance” then “Indexing Performance: Deployment”



#### Explore the IP:Deployment dashboard top panel

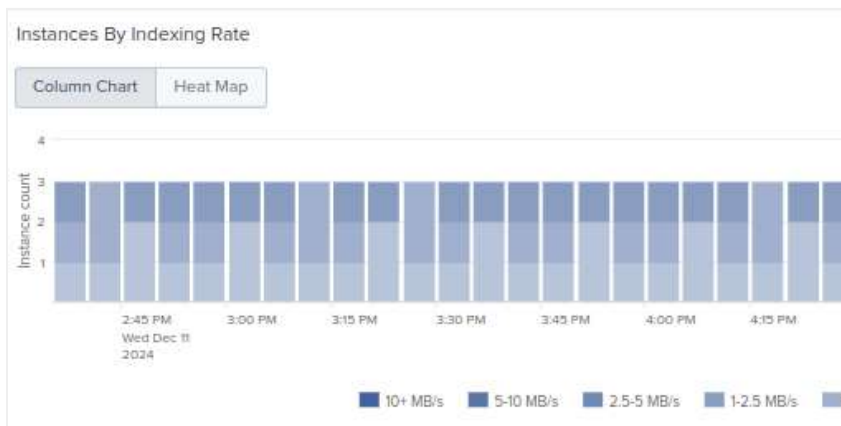


2. High level overview

- a. Pipeline Count
- b. Indexing rate
- c. Various Queue fill ratios

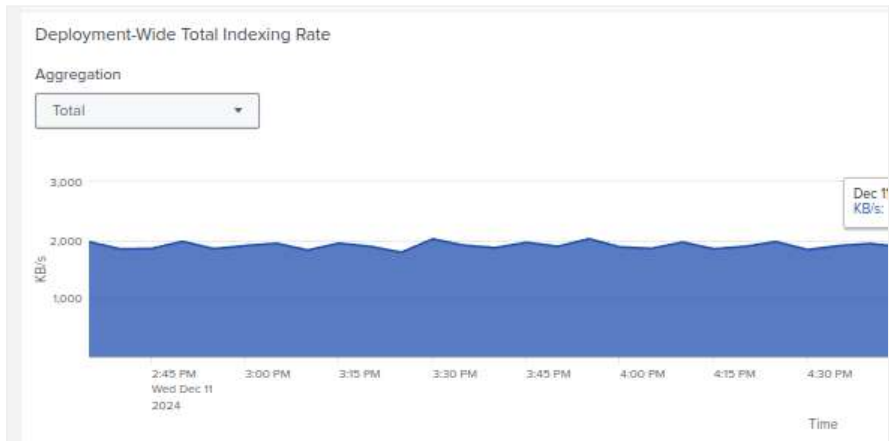
3. In the overview, we see some significant values, most relevant are the fill ratios. These are where we would detect the issues related most to ingestion on indexers. Looking at the fill rates, the “inbound” nature is read left to right, meaning that when an event enters the indexer processing sequence, the first queue it enters is the parsing queue. What that means is the to decide which queue is most likely causing issues, read right to left, and the first full queue needs to be addressed first, as it will likely be the root issue. Until this is cleared, other aspects of any performance problems are likely to be affected by the full queue. Here we see the Indexing queue is full, which likely means either resource starvation or slow disk. This will explored, but first, review the other panels.
4. Each of the dashboard elements are clickable and provide additional detail for the indexer

### Instances by Indexing Rate panel



5. provides information that can be used to better understand how the system is running. If the color is consistent, then the workload is evenly distributed, but if the color is different, there may be hotspots in the indexing systems.

## Deployment-Wide Indexing Rate panel



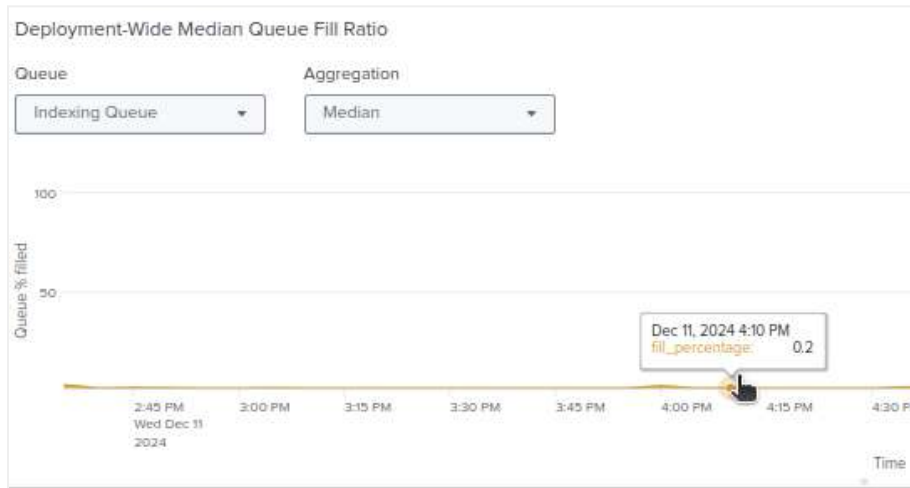
6. Provides context as to how much overall ingest is occurring within the environment.

## Instance by Median Queue Fill Ratio panel



7. Helps us understand the overall rate and where there may be differences that need to be addressed. The overall goal will usually be that the queue fill rates are at or near zero, but if they aren't this is potentially where attention must be paid on the indexers.

## Deployment-Wide Median Queue Fill Ratio panel



8. This panel provides context when used to find the most likely of the queues that are near full that may be causing ingestion slowdowns.

## Exercise 3 – Explore diagnosing slow indexing

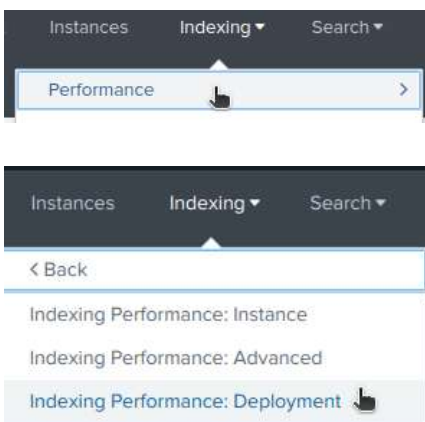
### Description

Examine the Overview and start diagnosing poor indexing performance through first the more detailed indexing performance dashboard and then OS tools.

### Steps

Use the already running GUI on the MC, access the IP:Deployment Dashboard

1. Select on the gray-bar “Indexing”, then “Performance” then “Deployment”



2. Because the s4a-ip-idx03 has the highest Indexing Ratio in the ‘Indexing Performance by Instance’ table, click on that row to go to the IP:Instance Dashboard

Indexing Performance by Instance							
3 Instances							
Instance	Pipeline Set Count	Indexing Rate (KB/s)	Status	Parsing Queue Fill Ratio (%)	Aggregation Queue Fill Ratio (%)	Typing Queue Fill Ratio (%)	Indexing Queue Fill Ratio (%)
s4a-ip-idx02	2	854	normal	pset0: 95.28 pset1: 0.00	pset0: 86.67 pset1: 0.00	pset0: 54.92 pset1: 0.00	pset0: 85.29 pset1: 0.04
s4a-ip-idx03	1	583	normal	99.17	99.96	99.95	100.00
s4a-ip-idx01	1	584	normal	83.25	57.82	70.87	66.32

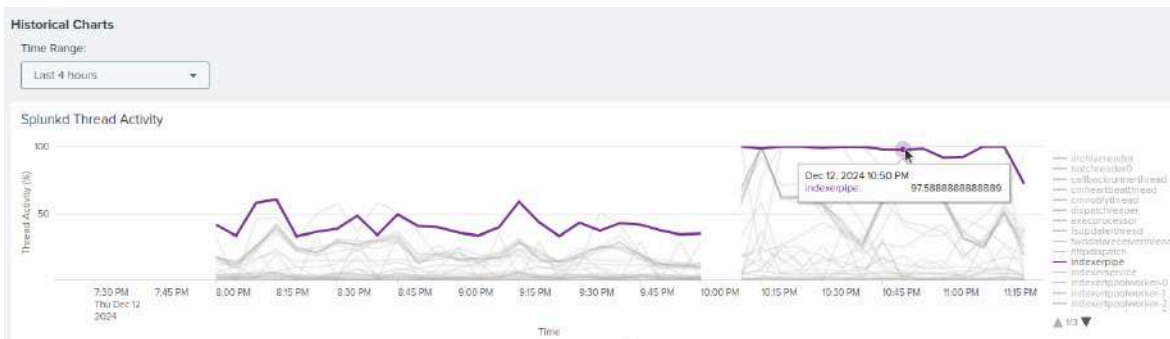
3. The high queue fill ratios should be read right to left, meaning the farthest right should be the first check. High indexing implies potential storage issues, so a comparison of the Indexing pipelines shows that even though the rate is higher elsewhere, the indexing queues are draining.

Indexing Performance by Instance							
3 Instances							
Instance	Pipeline Set Count	Indexing Rate (KB/s)	Status	Parsing Queue Fill Ratio (%)	Aggregation Queue Fill Ratio (%)	Typing Queue Fill Ratio (%)	Indexing Queue Fill Ratio (%)
s4a-ip-idx02	2	854	normal	pset0: 95.28 pset1: 0.00	pset0: 86.67 pset1: 0.00	pset0: 54.92 pset1: 0.00	pset0: 85.29 pset1: 0.04
s4a-ip-idx03	1	583	normal	99.17	99.96	99.95	100.00
s4a-ip-idx01	1	584	normal	83.25	57.82	70.87	66.32

- Given that each of these instances is identical, the behavior should be expected to be roughly similar. This tends to lead to a hypothesis that the issue is an underlying storage issue

**Go to the Indexer Performance:Advanced dashboard**

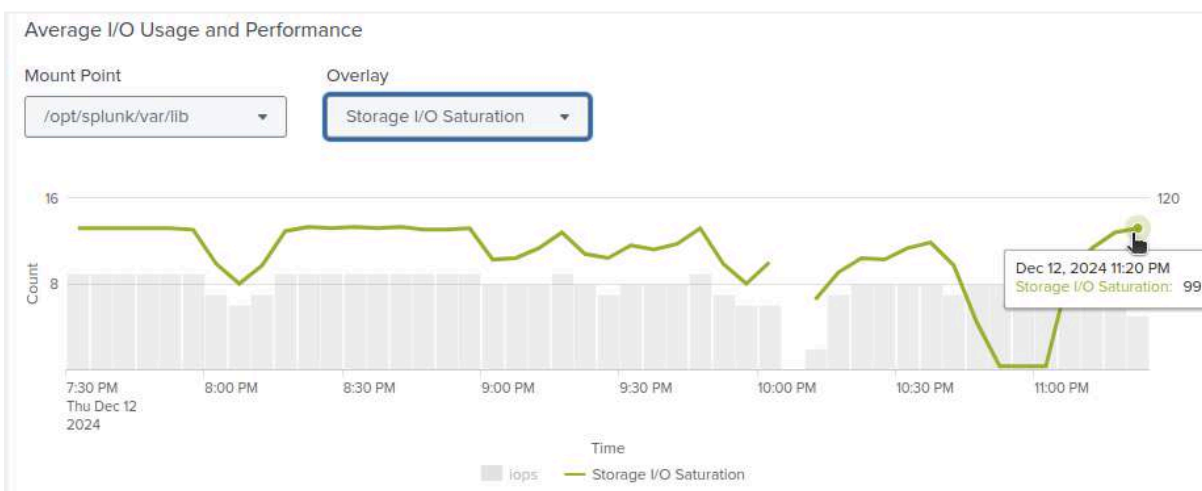
5. Select s4a-ip-idx03 will display, on the 'Historical Charts' section, the '*Splunkd Thread Activity*' will show which threads are most used -



- Shows that the 'indexerpipe' thread (and the batchreader) is busiest. This points to the culprit here being a slow storage environment.

**Go to the Resource Usage:Machine dashboard**

7. Select 's4a-ip-idx03' to continue the investigation, at the bottom of the page, on the 'Average I/O Usage and Performance' panel, select '/opt/splunk/var/lib', which is the mount point for the index storage on the indexers and then for the overlay, 'Storage I/O Saturation', and we see:



8. A high I/O Saturation, and a rather low IOPs Count (~8). To confirm the slow IO, checking at the OS level, using 'iostat' will be useful.

## Explore the issue from the OS

9. From the 'jumphost' login, ssh to s4a-ip-idx03

```
ssh s4a-ip-idx03
```

10. Find the device that the mount is using, we'll use the standard \*nix command 'df', but mount (and many others can be used)

```
df /opt/splunk/var/lib
```

```
splunk@s4a-ip-idx03:~$ df /opt/splunk/var/lib
Filesystem      1K-blocks      Used Available Use% Mounted on
/dev/sda         104806400 2981304 101825096   3% /opt/splunk/var/lib
```

11. Then execute 'iostat -xz 1'

```
iostat -xz 1
```

12. And see the results. The first batch of data is the statistics from boot, so not terribly useful. After that, extended statistics ('x') and compressed ('z') meaning only those disks with activity are shown, on a one second interval

sda	0.00	0.00	0.00	0.00	0.00	0.00	5.00	124.00	0.00	0.00	4381.80	24.86	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	21.91	100.00
avg-cpu:																						
	%user	%nice	%system	%iowait	%steal	%idle																
	34.12	0.00	4.58	8.59	0.00	32.71																
Device	r/s	rkB/s	rrqm/s	%rrqm	r_await	rrqm-sz	w/s	wkB/s	wrqm/s	%wrqm	w_await	wrqm-sz	d/s	dkB/s	drqm/s	%drqm	d_await	drqm-sz	f/s	f_await	aqu-sz	%util
dm-0	0.00	0.00	0.00	0.00	0.00	0.00	5.00	40.00	0.00	0.00	0.67	5.67	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	
sda	0.00	0.00	0.00	0.00	0.00	0.00	5.00	85.00	4.00	44.44	3315.20	17.20	0.00	0.00	0.00	0.00	0.00	0.00	1.00	8.00	16.35	100.00
sdb	0.00	0.00	0.00	0.00	0.00	0.00	5.00	40.00	1.00	16.67	1.00	8.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00
avg-cpu:																						
	%user	%nice	%system	%iowait	%steal	%idle																
	39.64	0.00	8.14	10.30	0.00	42.72																
Device	r/s	rkB/s	rrqm/s	%rrqm	r_await	rrqm-sz	w/s	wkB/s	wrqm/s	%wrqm	w_await	wrqm-sz	d/s	dkB/s	drqm/s	%drqm	d_await	drqm-sz	f/s	f_await	aqu-sz	%util
dm-0	0.00	0.00	0.00	0.00	0.00	0.00	89.00	859.50	0.00	0.00	3.69	9.66	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.33	2.40
sda	0.00	0.00	0.00	0.00	0.00	0.00	89.00	856.00	0.00	0.00	1396.00	11.20	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	6.98	100.00
sdb	0.00	0.00	0.00	0.00	0.00	0.00	69.00	859.50	20.00	22.87	2.58	12.46	0.00	0.00	0.00	0.00	0.00	0.00	2.00	2.50	0.18	2.60

13. This output supports the in-Splunk finding that the disk is the issue, and the interpretation here would be that because the "%util", which in the glances and top output as 'iowait' and 'wa', but across the entire system so lower percentages, is 100% for this device, and the 'w/s' is only 5, meaning that only 5 IOPs are occurring, that this is a slow disk.

a. Output from 'glances'

```
s4a-ip-idx03 (Ubuntu 22.04 64bit / Linux 5.15.0-125-generic)
Cloud c6a.12xlarge instance i-0436a206221630029 (us-east-1b)

QEMU Virtual CPU version 2.5+
CPU [|||||] 33.7%
MEM [|||||] 13.5%
SWAP [|||||] 0.0%

CPU \ 33.7% idle: 45.9% ctx_sw 2K
user 36.9% irq 0.0% inter 4K
system 9.9% nice 0.0% sw_int 2K
iowait 7.3% steal 0.0%
```

b. Output from 'top'

```
top - 23:55:22 up 1:45, 1 user, load average: 11.60, 11.46, 12.60
Tasks: 321 total, 3 running, 318 sleeping, 0 stopped, 0 zombie
%Cpu(s): 32.9 us, 11.9 sy, 0.0 ni, 44.2 id, 11.0 wa, 0.0 hi, 0.1 si, 0.0 st
MiB Mem : 11954.4 total, 4711.9 free, 1274.3 used, 5968.2 buff/cache
MiB Swap: 8188.0 total, 8188.0 free, 0.0 used, 10373.3 avail Mem
```

## Reset the nVM

14. This is born out, as the IOPs for the device in the nVM are being artificially restrained to provide something to discover! The IOPs are constrained to just 5, so iostat is returning the exact correct number of IOPs.

- From the 'jumphost', execute the command '*sudo virsh blkdeviotune s4a-ip-idx03 sdc --write\_iops\_sec 0*' to remove the constraint (yes, it is listed as sdc, but this is from the KVM side. It is confusing, but we do not control how KVM works!).

```
sudo virsh blkdeviotune s4a-ip-idx03 sdc --write_iops_sec 0
```

- This will remove the constraint.
- Go back and look at the iostat, glances, and top values. Then, in a few minutes, go back and look Average I/O Usage and Performance panel.



## Exercise 4 – Explore a node reporting full memory

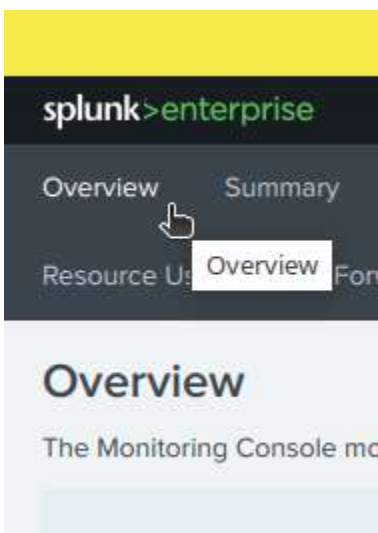
### Description

Examine the Overview and start diagnosing high-memory usage, follow through to the specifics related to the individual indexer, and assess the resource usage

### Steps

Use the already running GUI on the MC, access the Overview Dashboard

1. The initial, default dashboard -



2. Observe the Triggered Alerts

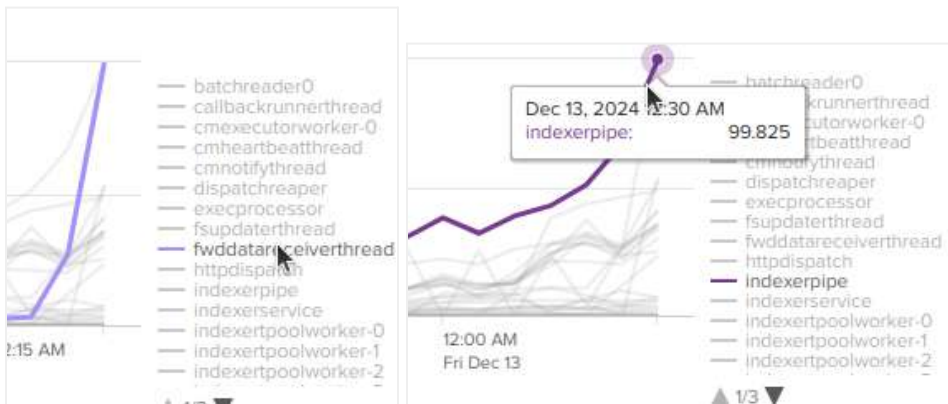
2 Triggered Alerts			
Enable or Disable		Manage triggered alerts	Filter by Last: 24 Hours ▾
			5 per page ▾
Alert Name ⇅	Instance ⇅	Time Triggered ⇅	Description ⇅
DMC Alert - Critical System Physical Memory Usage	s4a-ip-idx01	Dec. 13, 2024 12:23 AM	One or more instances has exceeded 90% memory usage.
DMC Alert - Saturated Event-Processing Queues	s4a-ip-idx01	Dec. 12, 2024 11:23 PM	One or more of your indexer queues is reporting a fill percentage, averaged over the last 15 minutes, of 90% or more.

We see that one of the nodes, s4a-ip-idx01 has alerted about memory

## Go to the IP:Advanced dashboard

- Open the **IP:Advanced dashboard**. Analyze Splunk performance - we saw that there was an error for Saturated Event Processing Queues on the overview, likely due, at least in part, to memory starvation.

On the *Splunkd Thread Activity* panel, we see that there seems to be an increase in CPU activity in both **fwddatareceiverthread** and the **indexerpipe** threads, so Splunk is definitely experiencing resource contention due to the RAM exhaustion



## Go to the Resource Usage:Deployment dashboard

- Open the **Resource Usage:Deployment** dashboard. Observe the RAM usage reported in the 'Resource Usage by Instance' panel - we see that the '*Physical Memory Usage (%)*' for s4a-ip-idx01 is quite high compared to the other indexers (~88% vs ~12%), so the hypothesis that issues are being caused RAM starvation may be real.

Resource Usage: Deployment

Role

All

Group

All

Hide Filters

Select views: All Snapshot Historical

Snapshots

Resource Usage by Instance

5 instance(s)

Instance ↕	Load Average ↕	CPU Cores (Physical / Virtual) ↕	CPU Usage (%) ↕	Physical Memory Capacity (MB) ↕	Physical Memory Usage (MB) ↕	Physical Memory Usage (%) ↕	I/O Operations per second (Mount Point) ↕	Storage I/O Saturation (Mount Point) ↕
s4a-ip-idx03	1.11	16 / 16	94.92	11954	1421	11.89	1 (/) 5 (/opt/splunk/var/lib)	0.19% (/) 99.36% (/opt/splunk/var/lib)
s4a-ip-idx02	0.80	16 / 16	90.73	11954	1504	12.58	12 (/opt/splunk/var/lib) 2 (/)	0.37% (/) 1.85% (/opt/splunk/var/lib)
s4a-ip-idx01	1.01	16 / 16	86.91	11954	10541	88.17	15 (/opt/splunk/var/lib) 6 (/)	1.97% (/) 4.59% (/opt/splunk/var/lib)
s4a-ip-mc	0.99	16 / 16	84.68	11954	1302	10.89	6 (/)	1.10% (/)
s4a-ip-c3m1	0.34	16 / 16	31.29	11954	831	6.95	3 (/)	0.77% (/)

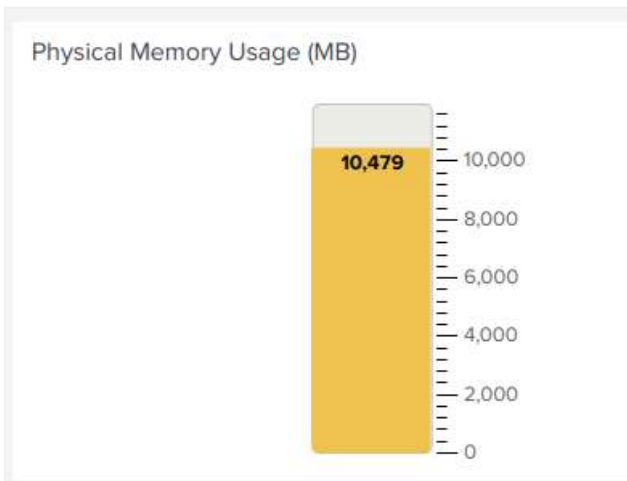
Click on instance name for more details.

Click on instance name for more details.

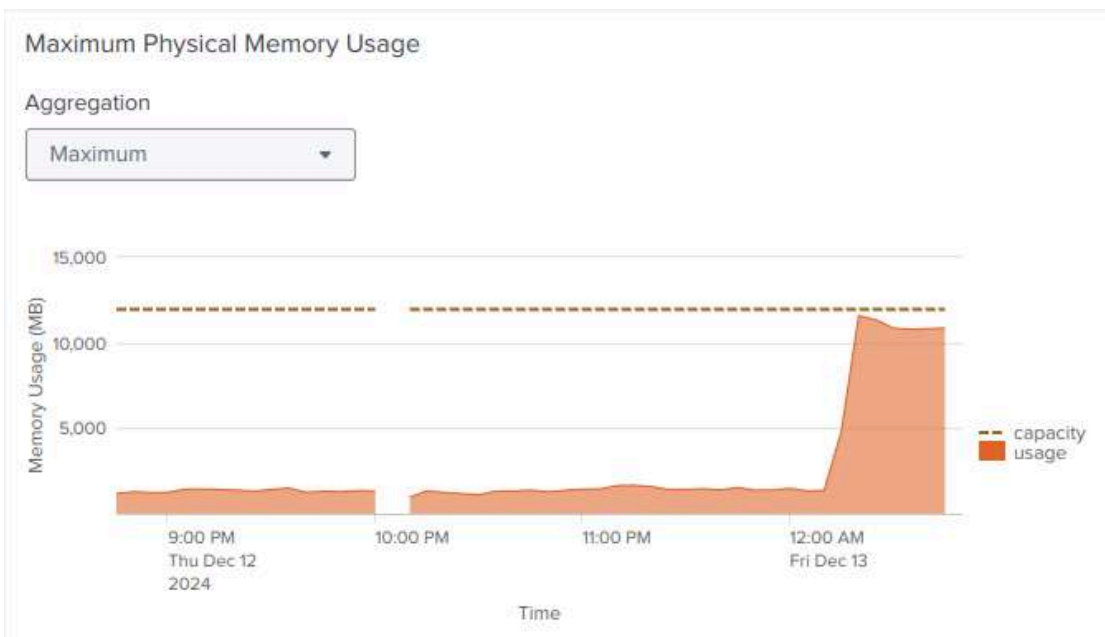
## Go to the Resource Usage:Machine dashboard

5. Open the **Resource Usage:Machine** dashboard. Be sure to select 's4a-ip-idx01' or it will be, by default, on the Cluster Manager, which is not what has been identified in the prior exploration as where the issue appears to be.

Note the 'Physical Memory Usage (MB)' panel



And also note the '<X> Physical Memory Usage' panel below in the historical charts, the dropdown can be used to select a different aggregation. Change the default 'Median' to 'Maximum', changing the presented data to the 'Maximum Physical Memory Usage'



This demonstrates that something is using RAM rather prodigiously. Time to diagnose this more thoroughly. Unfortunately, figuring out what is using RAM inside splunk is a bit more difficult, so

going to the OS is the best option. However, we have been able to identify that the culprit is s4a-ip-idx01, so back on the command line of the jumphost, ssh to s4a-ip-idx01

## Explore the issue from the OS

6. Access the indexer 01 command line with:

```
ssh -p 2222 attendee@s4a-ip-idx01
```

7. Bring up glances or top, here will use 'glances':

```
glances
```

```
s4a-ip-idx01 (Ubuntu 22.04 64bit / Linux 5.15.0-125-generic) Uptime: 3:07:
Cloud c6a.12xlarge instance i-0436a206221630029 (us-east-1b)

QEMU Virtual CPU version 2.5+
CPU 74.5% idle: 2.0% ctx_sw 3K MEM - 97.9% active 386M SWAP / 20.8% LOAD 16-cor
CPU [|||||] user 29.6% irq 0.0% inter 7K total 11.7G inactive 10.8G total 8.00G 1 min: 14.9
MEM [|||||] system 45.3% nice 0.0% sw_int 2K used 11.4G buffers 68K used 1.66G 5 min: 12.7
SWAP [|||||] iowait 18.8% steal 0.0% free 247M cached 367M free 6.34G 15 min: 14.1
```

Reviewing what glances shows us, we see that the system is, indeed, using a great deal of the RAM available.

The value of MEM at 97.6% implies that effectively all of the RAM is in use.

The SWAP column showing 1.66G of swap used supports the RAM oversubscription issue. The fact that more than a tiny amount of means that the kernel has had to remove unused memory pages into the disk cache. The competition for RAM, causing paging and/or swapping, will cause the entire system to begin to slow and show problems.

The 'iowait' callout provides a view as to what oversubscribed RAM can do, that indicates that, even with NVME disk, this instance is struggling to complete writes because it is competing for disk IO with the swap/page mechanism.

Another section of the glances screen shows us the processes that are running

CPU%	MEM%	VIRT	RES	PID	USER	TIME+	THR	NI	S	R/s	W/s	Command ('k' to kill)
74.6	83.4	11.0G	9.74G	88628	splunk	5:09	1	0	R	7M	0	tail
204	2.1	3.56G	254M	847	splunk	14h26:59	87	0	S	86K	1M	splunkd --under-systemd --systemd-del
99.9	1.4	476M	162M	90065	splunk	3:20	4	0	S	2M	2M	splunkd recover-metadata /opt/splunk/
61.9	0.5	364M	54.1M	62558	splunk	39:04	1	0	R	0	783	python3 /usr/bin/glances --theme-whit
0.2	0.4	1.79G	48.3M	58466	splunk	0:54	7	0	S	0	0	[splunkd pid=847] [search-launcher]
0.2	0.4	296M	42.3M	75103	splunk	0:15	8	0	S	0	0	[splunkd pid=847] [search-launcher]
0.2	0.3	290M	40.5M	75101	splunk	0:10	8	0	S	0	0	[splunkd pid=847] [search-launcher]
0.0	0.3	3.27G	39.3M	72531	splunk	0:36	8	0	S	0	0	[splunkd pid=847] [search-launcher]
0.0	0.3	4.81G	38.6M	69381	splunk	1:06	6	0	S	0	0	[splunkd pid=847] [search-launcher]
6.3	0.3	224M	38.6M	1434	splunk	15:37	13	0	S	0	0	splunkd instrument-resource-usage -p
0.0	0.3	3.27G	35.6M	72560	splunk	0:51	7	0	S	0	0	[splunkd pid=847] [search-launcher]

We can see that there is a tail process running that is using 9.74G of resident (live RAM). This is likely the culprit.

Not that the splunkd processes are NOT using too much RAM, but in your environment they would be expected to use much more, as they would be doing work, and they will likely be the culprit. If this is the case, then the path forward is to analyze the searches that are being executed to see if they are either using large lookups or returning exceptionally large data sets. If so, scale-out may not be the answer, there may be a requirement to increase the RAM on the indexers

The tail process is part of a script that is executing from rc.local, and can be terminated so see the system return to normal.

8. Use the command 'sudo systemctl stop rc-local' on s4a-ip-idx01 to remove the memory hog.

```
sudo systemctl stop rc-local
```

## Exercise 5 – Explore a node that is undersized for the workload

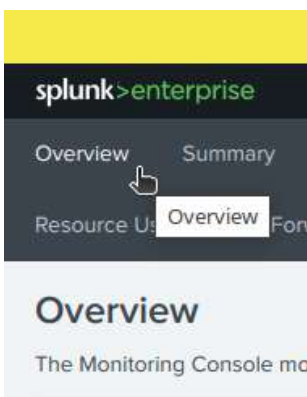
### Description

Examine the Overview and start diagnosing an under-resourced indexer node.

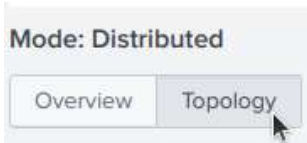
### Steps

Use the already running GUI on the MC, access the Overview dashboard

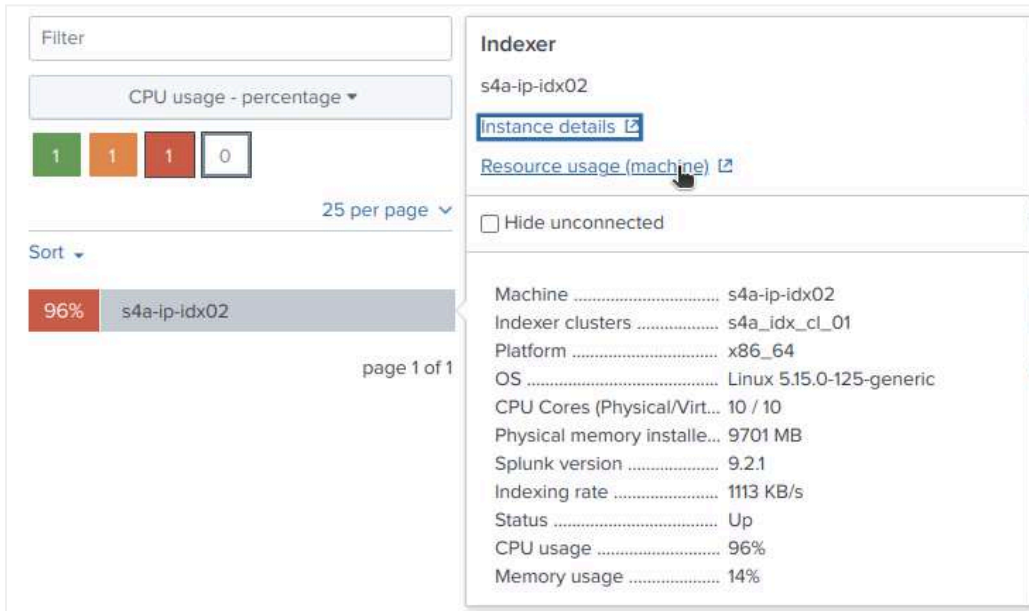
1. The initial, default dashboard -



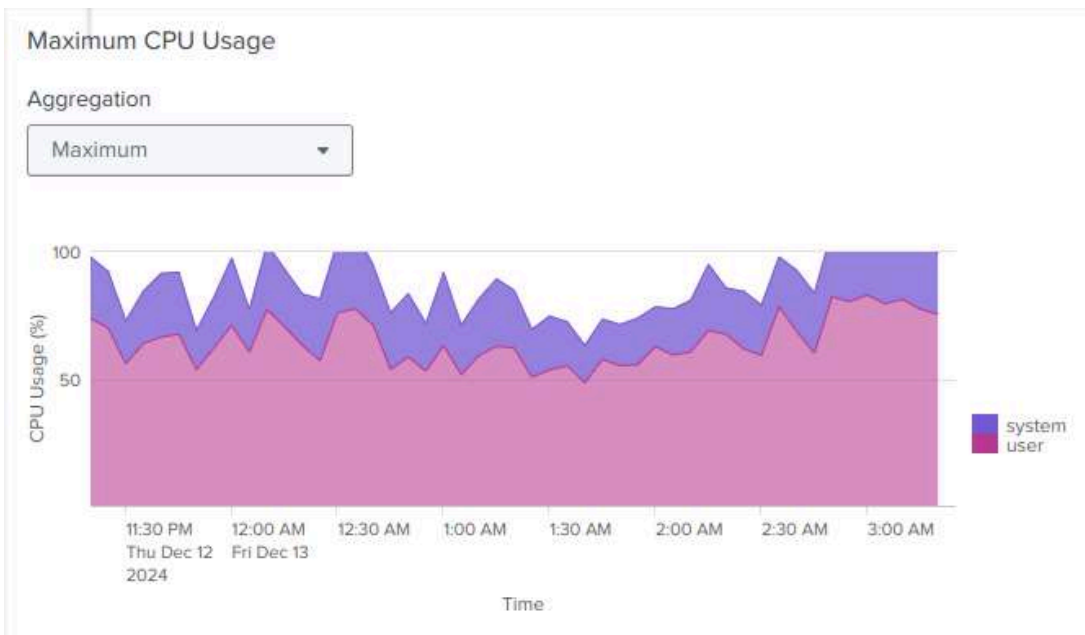
2. Select Topology



3. See the boxes. Then click the red box, and the node running hot will be displayed.



4. The three indexers are struggling, but one seems to be running extremely hot. It appears that s4a-ip-idx02 is using a great deal of the CPU resources.
5. Click on the 'Resource usage (machine)' link
6. The panel '<x> CPU Usage' is partway down the page. Change the 'Median' to 'Maximum'. This will show the running load on the system, over time.
7. Go the IP:Advanced dashboard



8. We see that the CPU usage is quite high. Time to investigate at the OS to confirm our hypothesis.

### Explore the issue from the OS

9. Access the command line via ssh:

```
ssh -p 2222 attendee@s4a-ip-idx02
```

10. Bring up top, here will use 'top'

```
top
```



```

top - 03:30:30 up 41 min, 1 user, load average: 9.73, 9.55, 10.32
Tasks: 268 total, 8 running, 260 sleeping, 0 stopped, 0 zombie
%Cpu(s): 76.1 us, 19.6 sy, 0.0 ni, 4.1 id, 0.0 wa, 0.0 hi, 0.3 si, 0.0 st
MiB Mem : 9701.9 total, 3555.6 free, 1374.3 used, 4772.0 buff/cache
MiB Swap: 8188.0 total, 8188.0 free, 0.0 used, 8042.3 avail Mem

```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
784	splunk	20	0	3600404	373136	75952	S	324.6	3.8	112:39.07	splunkd
21979	splunk	20	0	123376	48432	2740	R	96.2	0.5	1:02.75	splunk-optimize
22309	splunk	20	0	107900	11468	2540	R	91.4	0.1	0:10.14	splunk-optimize
22461	splunk	20	0	124328	50864	44576	R	47.6	0.5	0:01.49	splunkd
22463	splunk	20	0	126824	55984	48684	R	47.6	0.6	0:01.49	splunkd
22318	splunk	20	0	43316	38228	12560	S	40.9	0.4	0:04.93	python3.7
22468	splunk	20	0	123560	43944	38892	R	31.3	0.4	0:00.98	splunkd
22323	splunk	20	0	43192	38020	12348	S	23.0	0.4	0:05.17	python3.7
22322	splunk	20	0	43460	38260	12576	S	15.7	0.4	0:05.14	python3.7
1535	splunk	20	0	229828	76716	57976	S	12.1	0.8	2:41.86	splunkd
22482	splunk	20	0	7168	3580	2652	R	12.1	0.0	0:00.38	splunk-optimize
1423	splunk	20	0	585224	120864	35840	S	11.8	1.2	4:20.71	mongod
22487	splunk	20	0	78604	3328	2592	R	9.6	0.0	0:00.30	splunk-optimize
20515	splunk	20	0	70792	64324	15692	S	9.3	0.6	0:14.88	python3.7
22079	splunk	20	0	8056	3816	3024	R	3.2	0.0	0:01.88	top
22489	splunk	20	0	4332	2284	2044	S	2.6	0.0	0:00.08	sadc

11. Splunk is very good at optimizing workload, but when the workload pushes the available resources (here CPU) too far, overall performance suffers.
12. This is a 10CPU system (deliberately undersized) that demonstrates what a system that is under-resourced shows as.
13. The upper pink box shows the 'Load Average', meaning the number of blocked jobs waiting for access to the CPU. This number, for Splunk, should always be as low as possible, preferably below the CPU count (10). When the load average is high, and stays high (the averages are 1 minute, 5 minute, and 15 minute) the system has as much load as can be comfortably handled. This also means that any increase, in any way, will cause the system to begin to falter and slow down.

## Remediation

14. Either lower the workload (decrease search load or decrease ingest workload) or increase the CPUs available to the system. That would be easy for this node, as it is hosted on a physical (virtual ...) server with CPU resources available, but in the case of an actual physical host, that may be problematic.



## Exercise 6 – Explore network tools

### Description

Use a tools that can provide visibility of connections and throughput for network performance. Iftop is available for all of the major distributions from the repositories and is useful to get a feel for what the workload and connections are without getting the network team involved. It does not provide visibility into the routing or VLAN infrastructure.

MTR is available for all of the major distributions from the repositories and is used to understand another aspect of network performance, latency.

### Steps

#### Access the indexer 01 command line

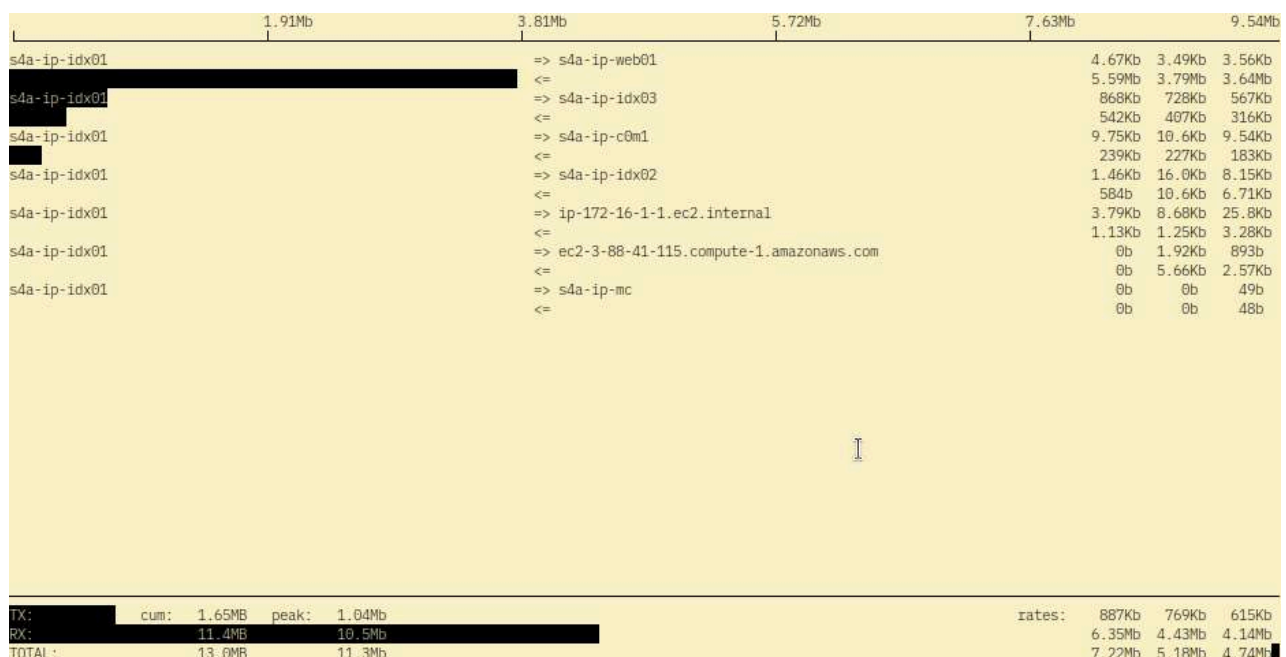
1. Access the indexer 01 command line with 'ssh -p 2222 attendee@s4a-ip-idx01

```
ssh -p 2222 attendee@s4a-ip-idx01
```

#### Engage iftop

2. Execute 'sudo iftop' - iftop accesses network kernel registers which requires enhanced privileges

```
sudo iftop
```



3. The connections, with a bar indicating overall throughput, both outbound and inbound down the left side. The right side, opposite the connection information provides 2, 10, and 40 second throughput rates.
4. Hitting “S” provides ports on the connections of the left.
5. The bottom of the screen provides summary statistics, with the columns on the right depicting, as above, 2, 10, and 40 second summary statistics.

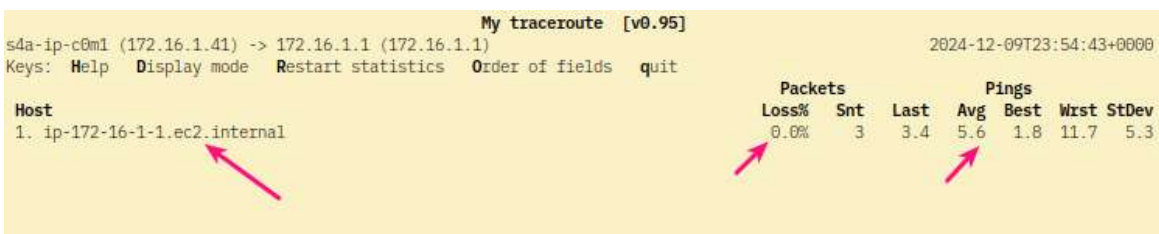
### Access the Cluster Manager command line

6. Access the cluster manager command line with ‘ssh -p 2222 attendee@s4a-ip-c0m1

```
ssh -p 2222 attendee@s4a-ip-c0m1
```

### Engage mtr

7. Execute ‘mtr s4a-ip-idx01 to see pathing, packet loss, and packet loss



Host		Packets		Pings				
		Loss%	Snt	Last	Avg	Best	Worst	StDev
1.	ip-172-16-1-1.ec2.internal	0.0%	3	3.4	5.6	1.8	11.7	5.3

8. This provides the destination, the packet loss rate, and the ICMP ping latency. Obviously, ICMP is not how most of the traffic will traverse the environment (TCP ...), but it does provide an effective view of up to layer 3 network traffic. If this number is above 100ms for inter-indexer/cluster manager traffic, the performance of all of the indexers will be poor.

## Links

Title	Link
Maximizing performance with the latest Splunk platform capabilities	<a href="https://lantern.splunk.com/Splunk_Platform/UCE/Observability/Maximizing_performance_with_the_latest_Splunk_platform_capabilities">https://lantern.splunk.com/Splunk_Platform/UCE/Observability/Maximizing_performance_with_the_latest_Splunk_platform_capabilities</a>
Planning for infrastructure and resource scalability	<a href="https://lantern.splunk.com/Splunk_Platform/Splunk_Outcome_Paths/Improve_Performance/Planning_for_infrastructure_and_resource_scalability">https://lantern.splunk.com/Splunk_Platform/Splunk_Outcome_Paths/Improve_Performance/Planning_for_infrastructure_and_resource_scalability</a>
Splunk Validated Architectures	<a href="https://docs.splunk.com/Documentation/SVA/current/Architectures/About">https://docs.splunk.com/Documentation/SVA/current/Architectures/About</a>
Reference hardware	<a href="https://docs.splunk.com/Documentation/Splunk/latest/Capacity/Referencehardware">https://docs.splunk.com/Documentation/Splunk/latest/Capacity/Referencehardware</a>
Indexing and search architecture	<a href="https://lantern.splunk.com/Splunk_Success_Framework/Platform_Management/Indexing_and_search_architecture">https://lantern.splunk.com/Splunk_Success_Framework/Platform_Management/Indexing_and_search_architecture</a>
Configure parallel reduce search processing	<a href="https://docs.splunk.com/Documentation/Splunk/latest/DistSearch/Setupparallelreduce">https://docs.splunk.com/Documentation/Splunk/latest/DistSearch/Setupparallelreduce</a>
Capacity Planning Manual	<a href="https://docs.splunk.com/Documentation/Splunk/latest/Capacity/Performancechecklist">https://docs.splunk.com/Documentation/Splunk/latest/Capacity/Performancechecklist</a>
Summary of performance recommendations	<a href="https://docs.splunk.com/Documentation/Splunk/latest/Capacity/Summaryofperformancerecommendations">https://docs.splunk.com/Documentation/Splunk/latest/Capacity/Summaryofperformancerecommendations</a>
Sizing your Splunk architecture	<a href="https://lantern.splunk.com/Splunk_Platform/Product_Tips/Administration/Sizing_your_Splunk_architecture">https://lantern.splunk.com/Splunk_Platform/Product_Tips/Administration/Sizing_your_Splunk_architecture</a>
What storage type should I use for a role?	<a href="https://docs.splunk.com/Documentation/Splunk/latest/Capacity/Referencehardware">https://docs.splunk.com/Documentation/Splunk/latest/Capacity/Referencehardware</a>
Indexer Memory Specification	<a href="https://docs.splunk.com/Documentation/Splunk/latest/Capacity/Referencehardware#:~:text=search%20performance.-,Indexer,-When%20you%20distribute">https://docs.splunk.com/Documentation/Splunk/latest/Capacity/Referencehardware#:~:text=search%20performance.-,Indexer,-When%20you%20distribute</a>
Network latency limits for clustered deployments	<a href="https://docs.splunk.com/Documentation/Splunk/latest/Capacity/Referencehardware#:~:text=performance%20recommendations.-,Network%20latency%20limits%20for%20clustered%20deployments,-A%20Splunk%20environment">https://docs.splunk.com/Documentation/Splunk/latest/Capacity/Referencehardware#:~:text=performance%20recommendations.-,Network%20latency%20limits%20for%20clustered%20deployments,-A%20Splunk%20environment</a>
How search types affect Splunk Enterprise performance	<a href="https://docs.splunk.com/Documentation/Splunk/latest/Capacity/HowsearchtypesaffectSplunkEnterpriseperformance">https://docs.splunk.com/Documentation/Splunk/latest/Capacity/HowsearchtypesaffectSplunkEnterpriseperformance</a>
How concurrent users and searches impact performance	<a href="https://docs.splunk.com/Documentation/Splunk/latest/Capacity/Accommodatemany simultaneous searches">https://docs.splunk.com/Documentation/Splunk/latest/Capacity/Accommodatemany simultaneous searches</a>
Reducing search load	<a href="https://lantern.splunk.com/Splunk_Platform/Splunk_Outcome_Paths/Reduce_Costs/Reducing_search_load">https://lantern.splunk.com/Splunk_Platform/Splunk_Outcome_Paths/Reduce_Costs/Reducing_search_load</a>
Scalability in IT: The Complete Guide To Scaling	<a href="https://www.splunk.com/en_us/blog/learn/scalability.html">https://www.splunk.com/en_us/blog/learn/scalability.html</a>
NixCraft System Monitoring Tools	<a href="https://www.cyberciti.biz/tips/top-linux-monitoring-tools.html">https://www.cyberciti.biz/tips/top-linux-monitoring-tools.html</a>
Monitoring Splunk Enterprise	<a href="https://docs.splunk.com/Documentation/Splunk/latest/DMC/DMCoverview">https://docs.splunk.com/Documentation/Splunk/latest/DMC/DMCoverview</a>

Glances	<a href="https://github.com/nicolargo/glances">https://github.com/nicolargo/glances</a>
Linux interface analytics on-demand with iftop	<a href="https://www.redhat.com/en/blog/linux-interface-iftop">https://www.redhat.com/en/blog/linux-interface-iftop</a>
Resource Usage	<a href="https://docs.splunk.com/Documentation/Splunk/latest/DMC/ResourceusageDeployment">https://docs.splunk.com/Documentation/Splunk/latest/DMC/ResourceusageDeployment</a>
top – Process activity monitoring command	<a href="https://www.cyberciti.biz/tips/top-linux-monitoring-tools.html#:~:text=top%20%E2%80%93%20Process%20activity%20monitoring%20command">https://www.cyberciti.biz/tips/top-linux-monitoring-tools.html#:~:text=top%20%E2%80%93%20Process%20activity%20monitoring%20command</a>
Brendan Gregg: Linux Performance	<a href="https://www.brendangregg.com/linuxperf.html">https://www.brendangregg.com/linuxperf.html</a>
Linux troubleshooting 101: System performance	<a href="https://www.redhat.com/en/blog/troubleshooting-system-performance">https://www.redhat.com/en/blog/troubleshooting-system-performance</a>
iostat – Monitor Linux disk activity	<a href="https://www.cyberciti.biz/tips/top-linux-monitoring-tools.html#:~:text=7.-,iostat,-%E2%80%93%20Monitor%20Linux%20average">https://www.cyberciti.biz/tips/top-linux-monitoring-tools.html#:~:text=7.-,iostat,-%E2%80%93%20Monitor%20Linux%20average</a>
mtr - monitor network connections and latency	<a href="https://www.cyberciti.biz/tips/finding-out-a-bad-or-simply-overloaded-network-link-with-linuxunix-os-es.html">https://www.cyberciti.biz/tips/finding-out-a-bad-or-simply-overloaded-network-link-with-linuxunix-os-es.html</a>
Indexing: Inputs: HTTP Event Collector	<a href="https://docs.splunk.com/Documentation/Splunk/latest/DMC/Inputdashboards">https://docs.splunk.com/Documentation/Splunk/latest/DMC/Inputdashboards</a>
Indexing: Inputs: Data Quality	<a href="https://docs.splunk.com/Documentation/Splunk/latest/DMC/Dataquality">https://docs.splunk.com/Documentation/Splunk/latest/DMC/Dataquality</a>
What is SSH (Secure Shell)?	<a href="https://www.ssh.com/academy/ssh">https://www.ssh.com/academy/ssh</a>
Download PuTTY: latest release	<a href="https://putty.org/">https://putty.org/</a>