Workload Management

To Achieve Efficient Resource Utilization

Bharath Aleti Product Management, Splunk
Anish Shrigondekar Principal Software Engineer, Splunk

October 2018

Forward-Looking Statements

During the course of this presentation, we may make forward-looking statements regarding future events or the expected performance 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 could differ materially. For important factors that may cause actual results to differ from those contained in our forward-looking statements, please review our filings with the SEC.

The forward-looking statements made in this presentation are being made as of the time and date of its live presentation. If reviewed after its live presentation, this presentation may not contain current or accurate information. We do not assume any obligation to update any forward-looking statements we may make. 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 functionality described or to include any such feature or functionality in a future release.

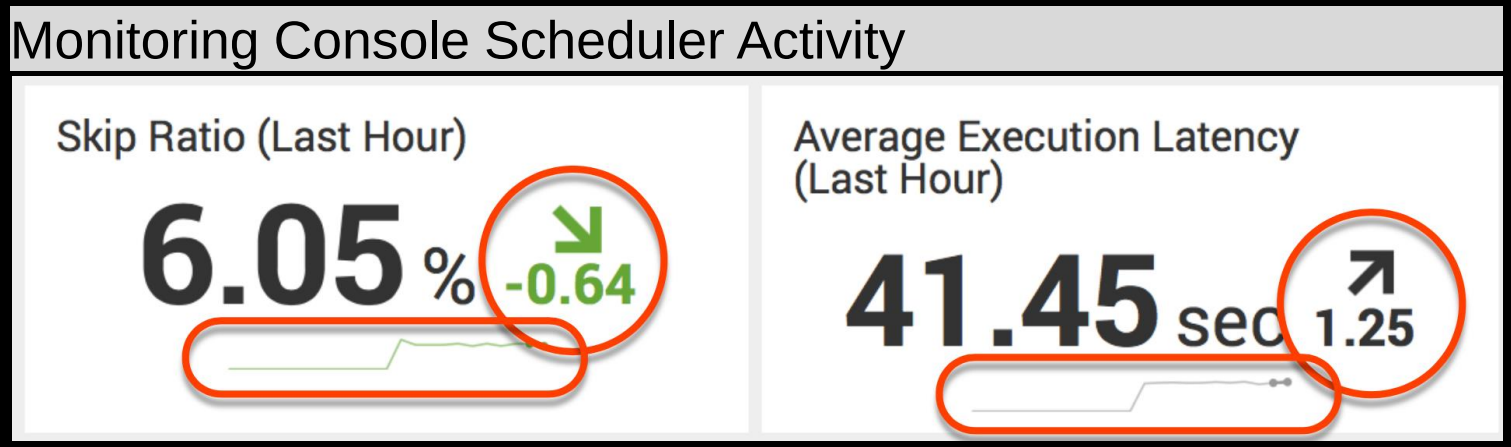
splunk> .conf18

Workload Management

How does it help ?

Challenges

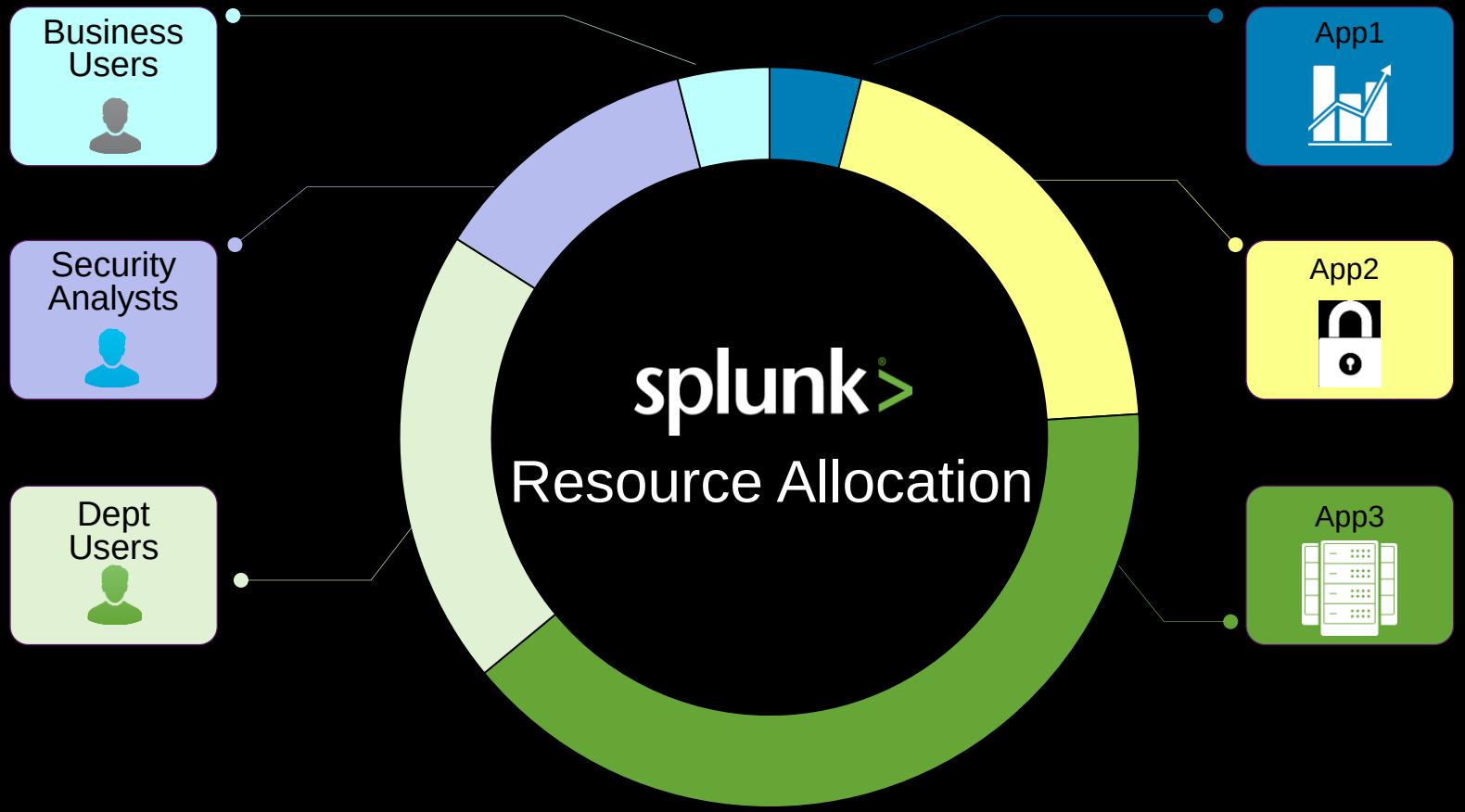
Unpredictability in search execution



- New users run wildcard searches, all-time searches, real time searches
- Users/apps running wildcard or expensive searches
- Resource utilization goes through the roof and impacts other users
- High search load impacts data ingestion lag and fails to fire alerts in time

Resource Allocation

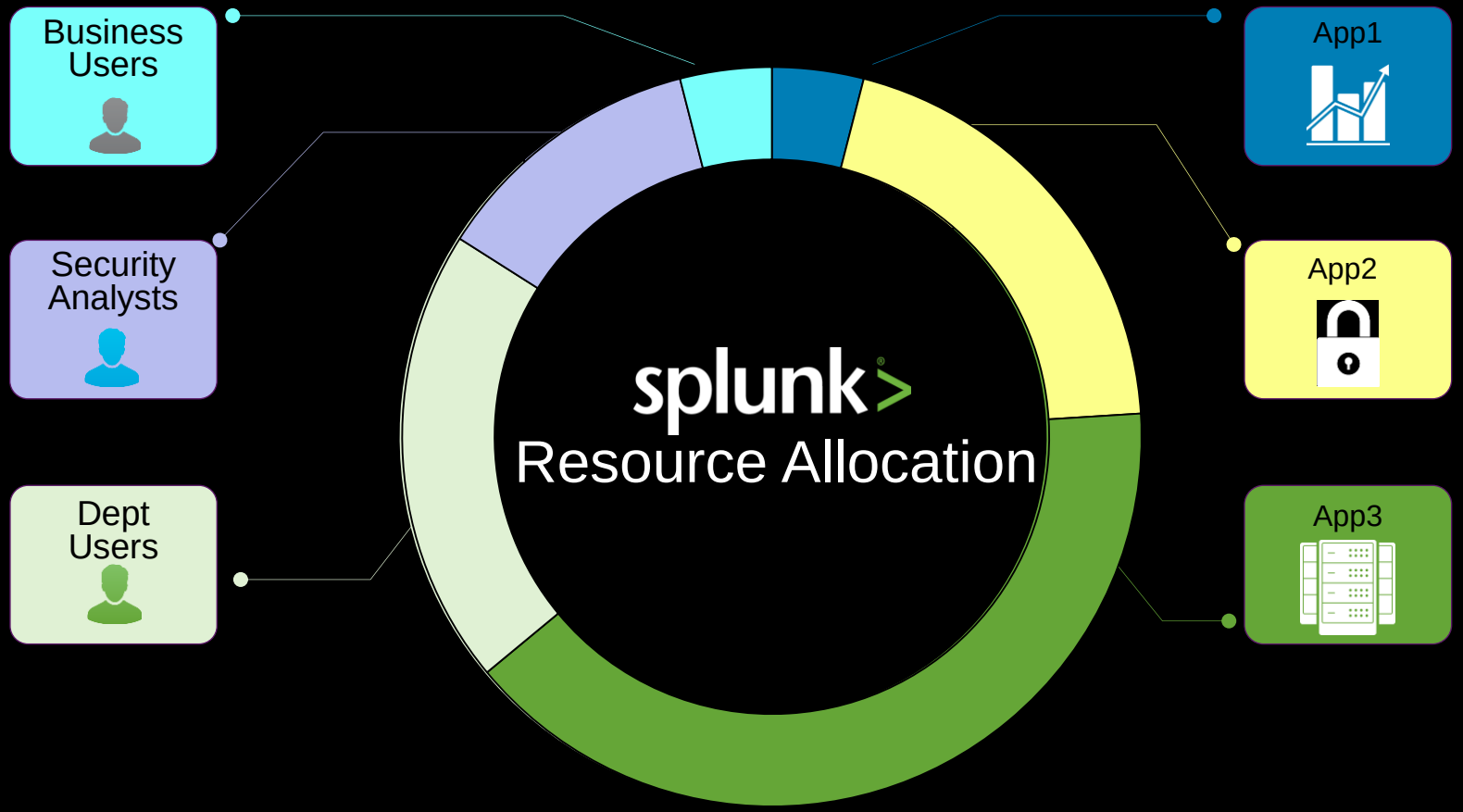
Lack of control to align resource allocation with business priorities



App Priority	User Priority
App1	Business users
App2	Security Analysts
App3	Dept Users

Resource Contention

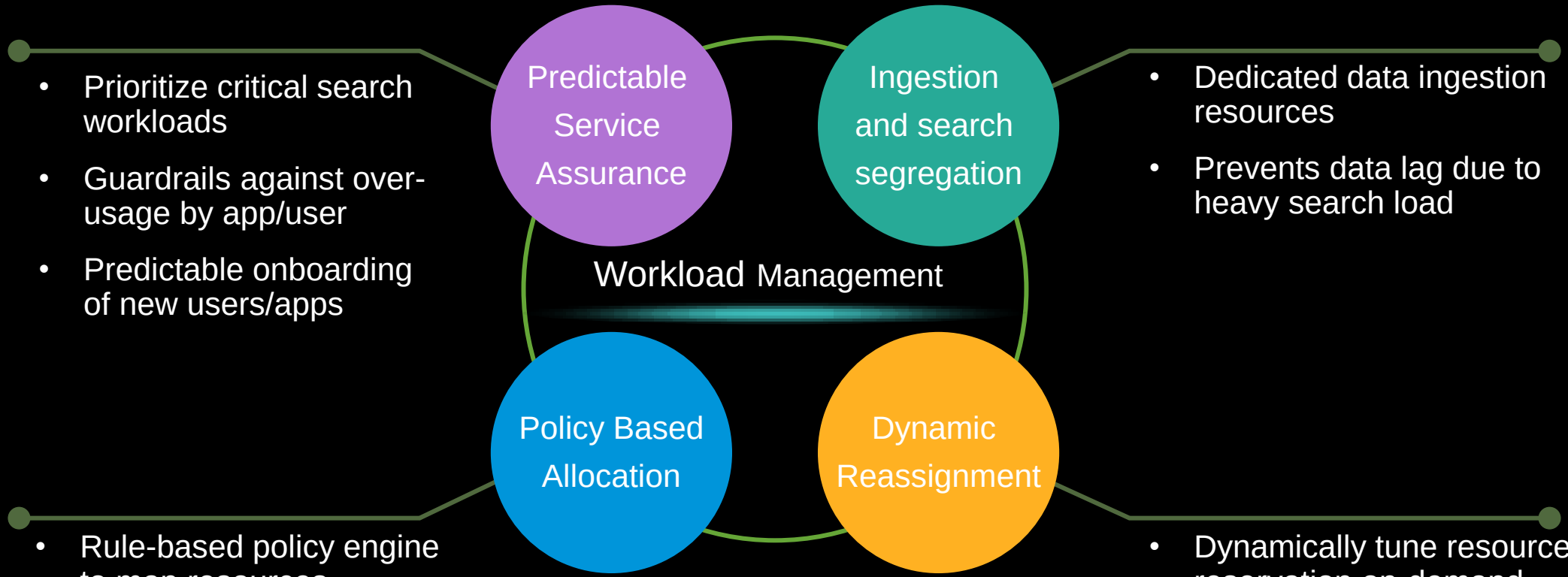
No guardrails over excessive resource usage



App Priority	User Priority
App1	Business users
App2	Security Analysts
App3	Dept Users

Workload Management

Aligns resource allocation with business priorities



- Prioritize critical search workloads
- Guardrails against over-usage by app/user
- Predictable onboarding of new users/apps

- Dedicated data ingestion resources
- Prevents data lag due to heavy search load

- Rule-based policy engine to map resources
- Resource assignment based on apps/roles

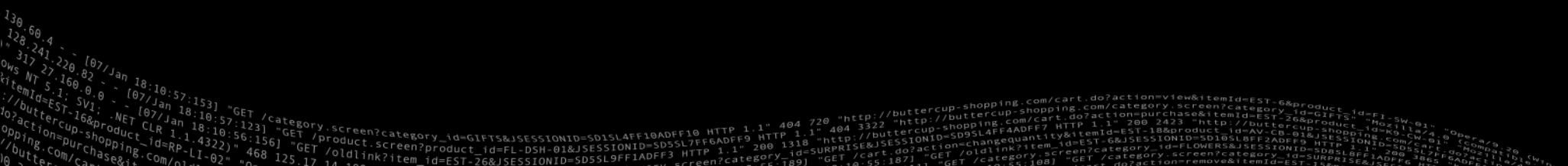
- Dynamically tune resource reservation on-demand
- Admin delegation through capabilities

Workload Pools and Rules

Policy Based Assignment of Resources

Workload Pools	CPU	Memory	Workload Rules/Policy
Search High	35%	35%	App=App1 Role=BusinessUser
Search Medium	20%	20%	App=App2 Role=SecAnalyst
Search Low	15%	15%	App=App3 Role=DeptUsers
Default Ingest	30%	30%	

App Priority	User Priority
App1	Business users
App2	Security Analysts
App3	Dept Users



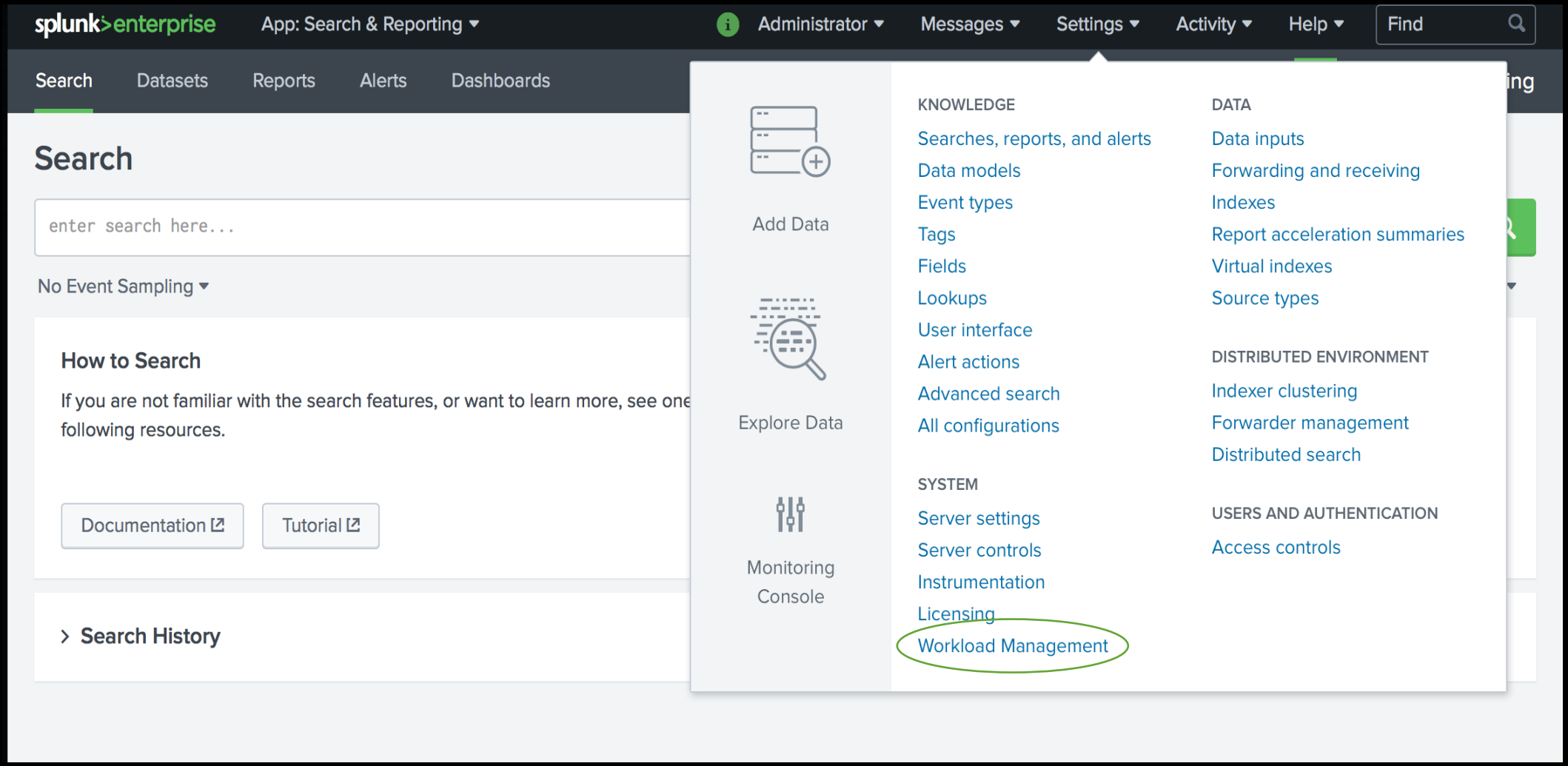
Workload Management

Control over resource allocation and enforced guardrails

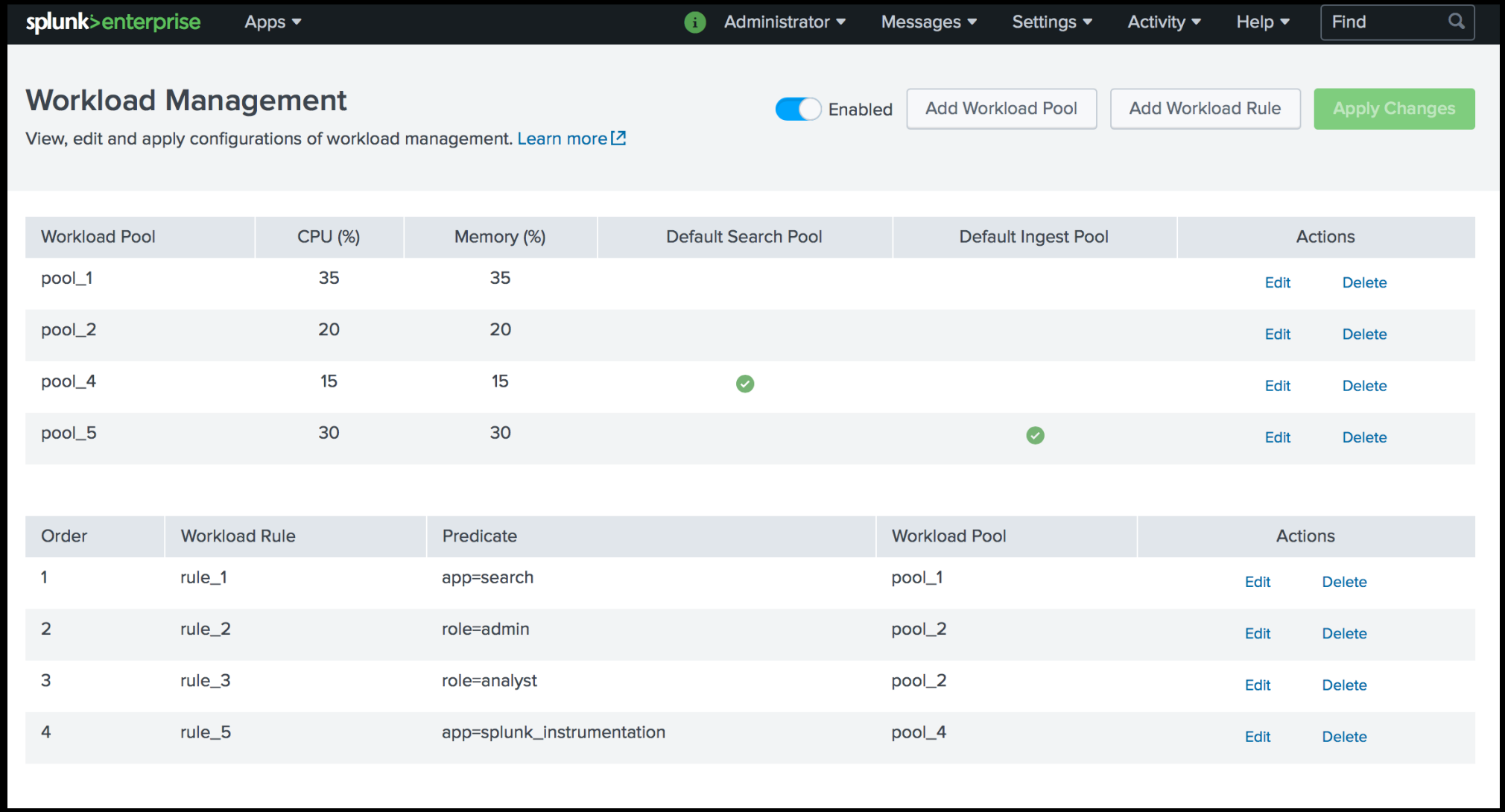
App Priority	User Priority
App1	Business users
App2	Security Analysts
App3	Dept Users

Workload Pools	CPU	Memory	Workload Rules/Policy
Search High	35%	35%	App=App1 Role=BusinessUser
Search Medium	20%	20%	App=App2 Role=SecAnalyst
Search Low	15%	15%	App=App3 Role=DeptUsers
Default Ingest	30%	30%	

Workload Management (Admin)



Workload Management (Admin)



Configuration Files

workload_pools.conf

```
[workload_pool:pool_1]
cpu_weight = 25
mem_weight = 25

[workload_pool:pool_2]
cpu_weight = 20
mem_weight = 20

[workload_pool:pool_4]
cpu_weight = 15
mem_weight = 15

[workload_pool:pool_5]
cpu_weight = 30
mem_weight = 30

[general]
default_pool = pool_4
enabled = 1
ingest_pool = pool_5
```

workload_rules.conf

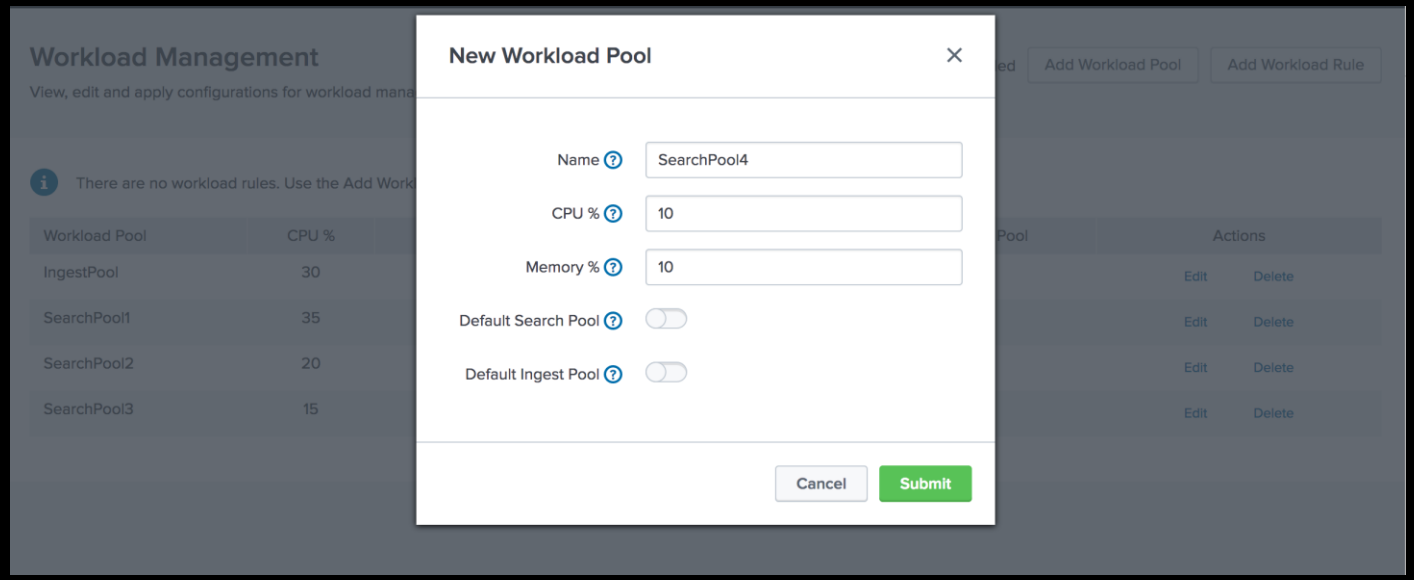
```
[workload_rule:rule_1]
predicate = app=search
workload_pool = pool_1

[workload_rule:rule_2]
predicate = role=admin
workload_pool = pool_2

[workload_rule:rule_3]
predicate = role=analyst
workload_pool = pool_2

[workload_rules_order]
rules = workload_rule:rule_1,workload_rule:rule_2,workload_rule
:rule_3,workload_rule:rule_5
rules_number = 4

[workload_rule:rule_5]
predicate = app=splunk_instrumentation
workload_pool = pool_4
```

splunk> .conf18

Workload Management: Pool Creation

Workload Pools

- CPU/Memory resource pools
- Specify allocation of CPU and memory resources
- Indicate ingest or search pool



First Version

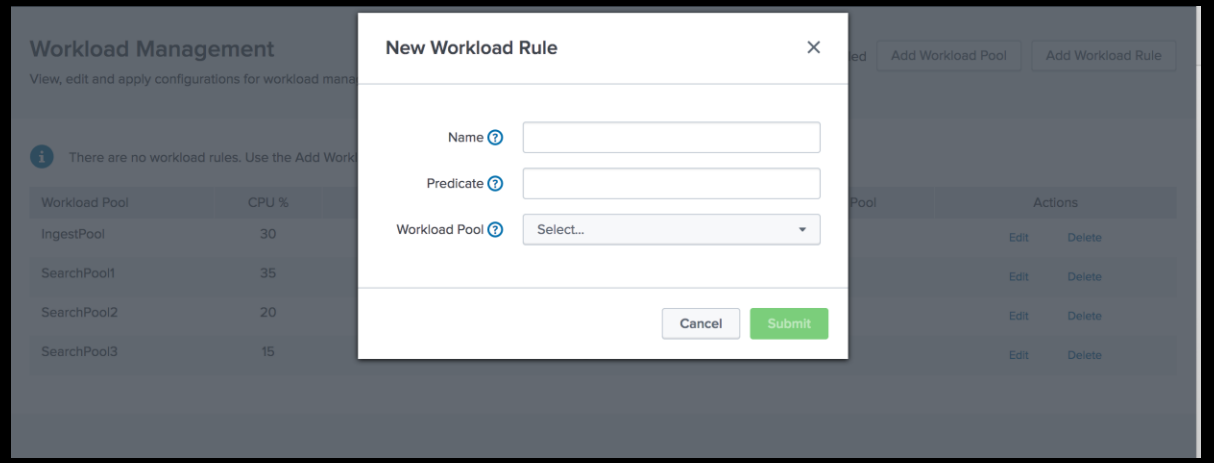
- One ingest pool
- Multiple Search Pools

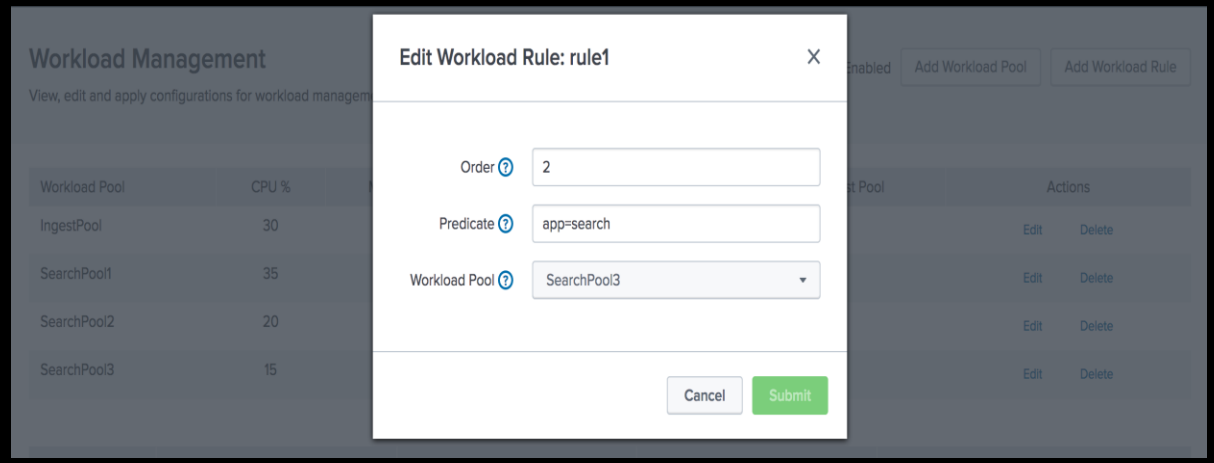
splunk> .conf18

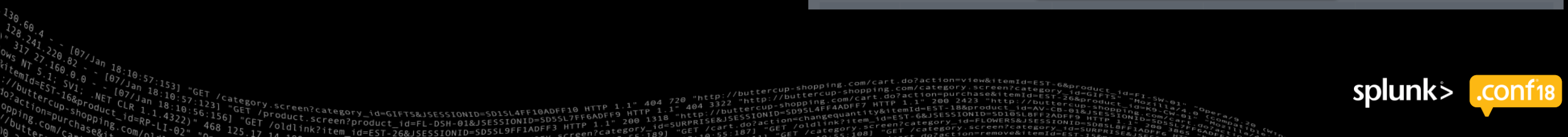
Workload Management: Rule Creation

Workload Rules:

- Determines assignment of pools based on custom rules
- Rules can be specified on an app, role or user basis
- Rules order determines rule precedence
- Provision to provide more complex rules going forward



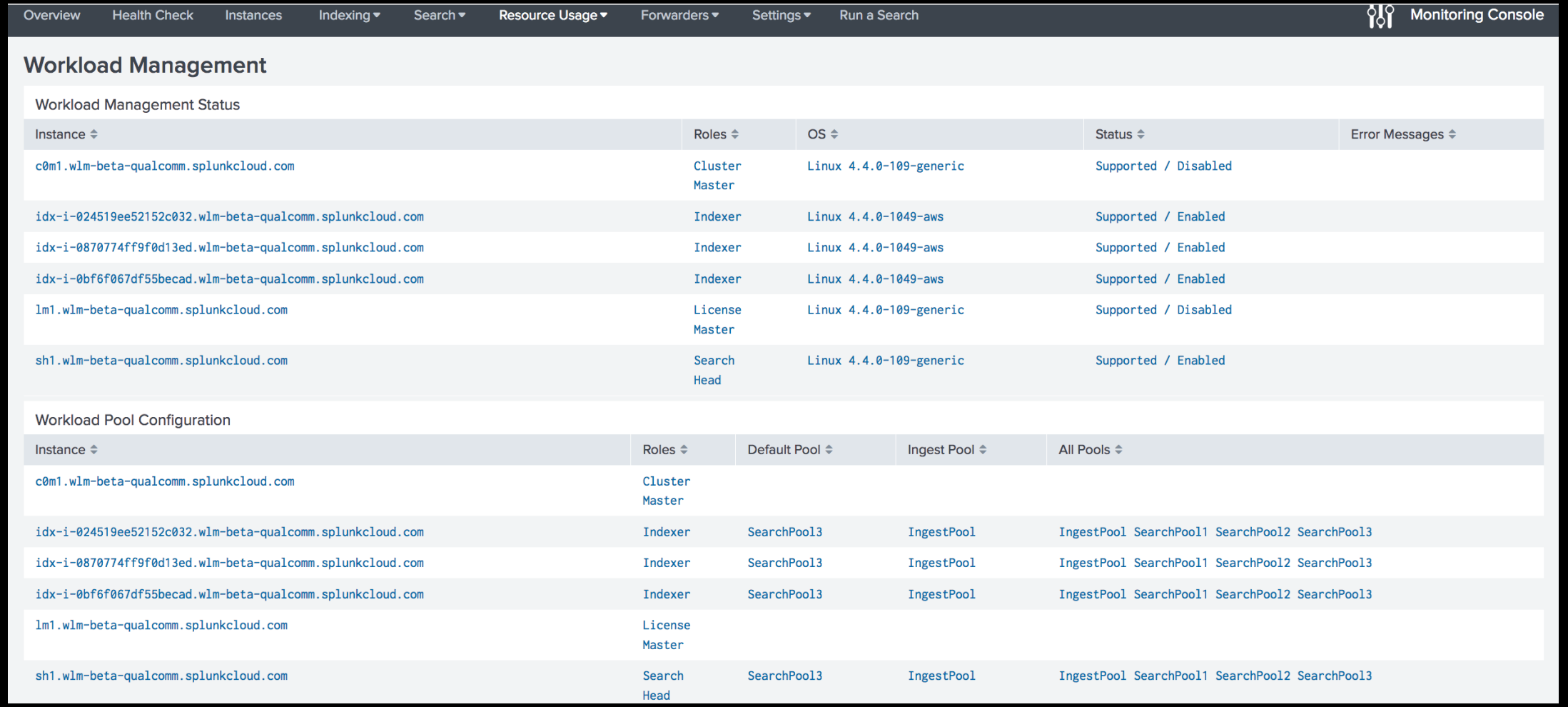




splunk> .conf18

Monitoring

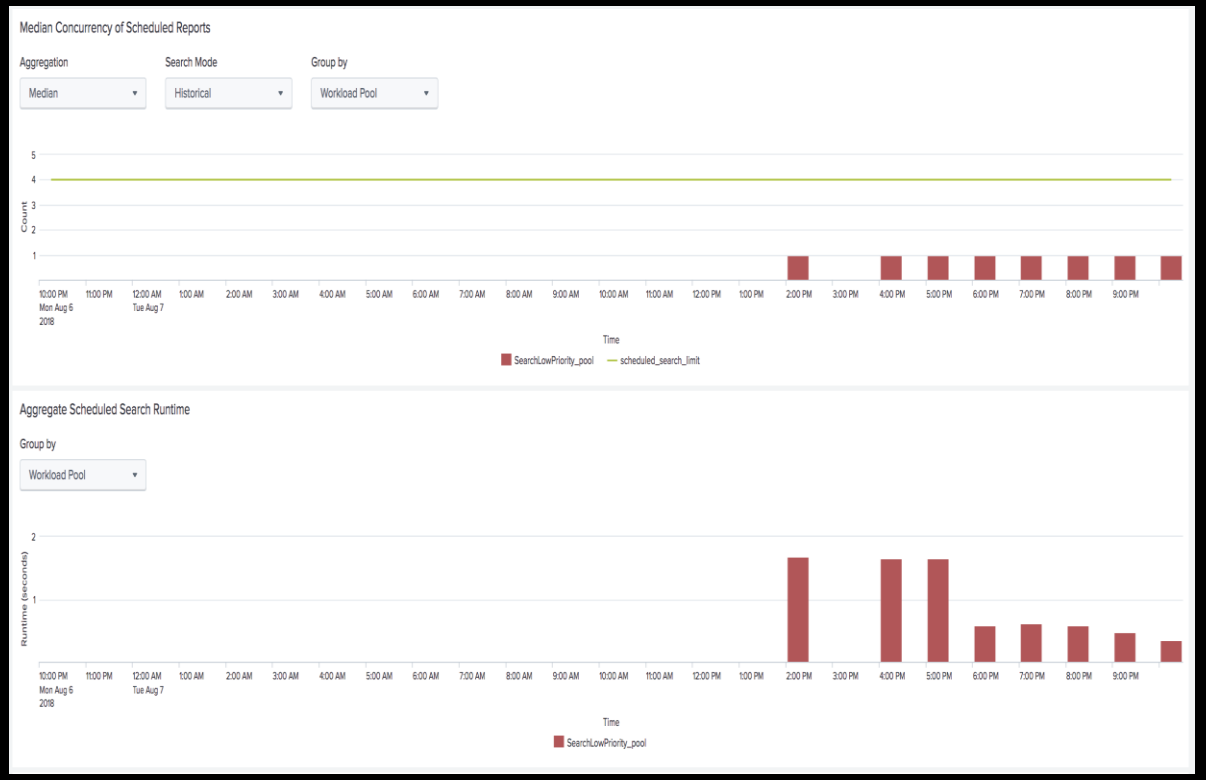
Monitor workload management status and resource usage



splunk> .conf18

Monitor Search and Scheduler Activity

Using New Workload Pools Filter



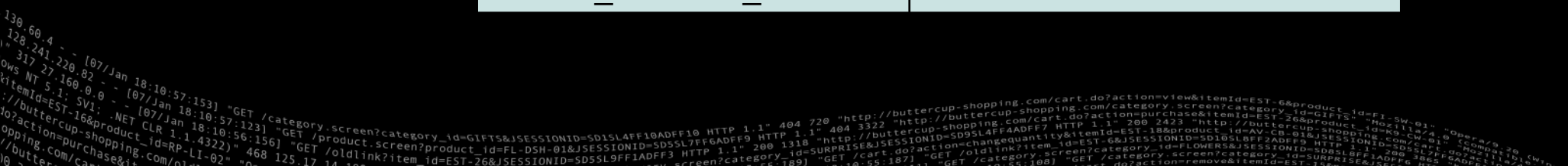
splunk> .conf18

Dynamic Reassignment

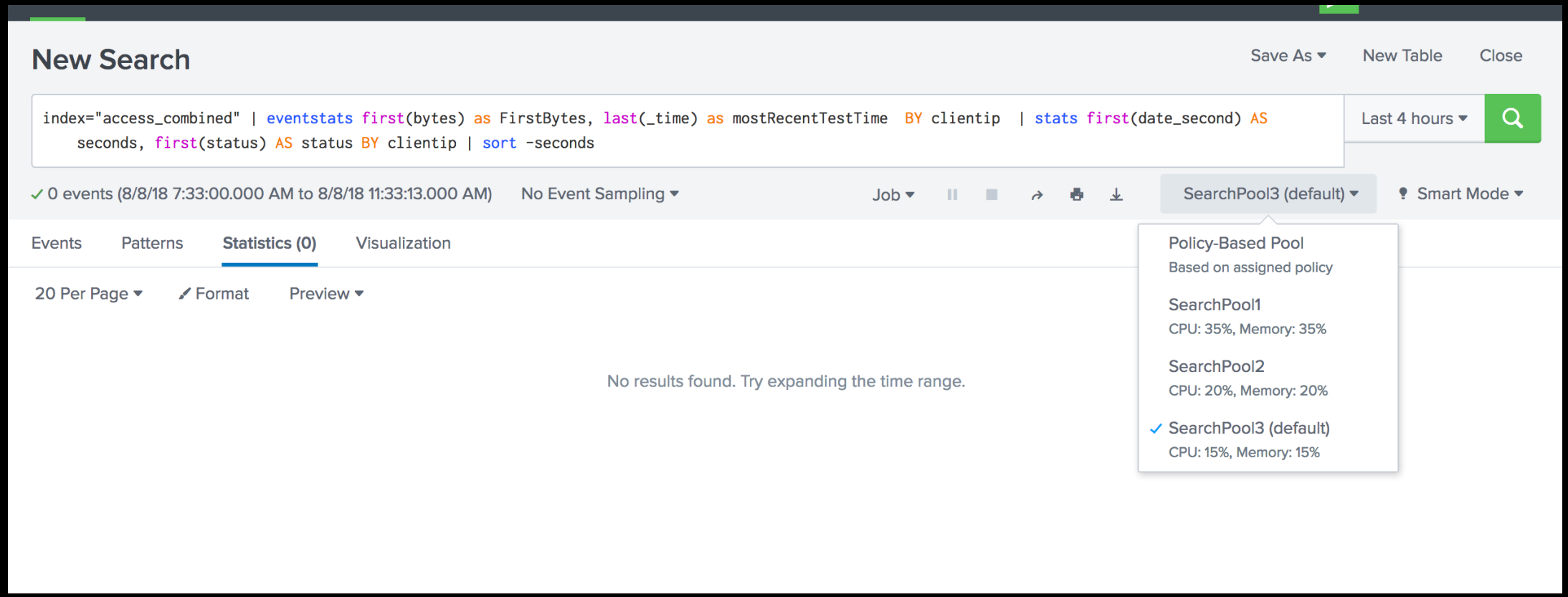
On-demand Reassignment of workload pool

- Available through capabilities
- Allows power users to use higher/lower priority workload pools on-demand
- Available to ad-hoc and scheduled searches

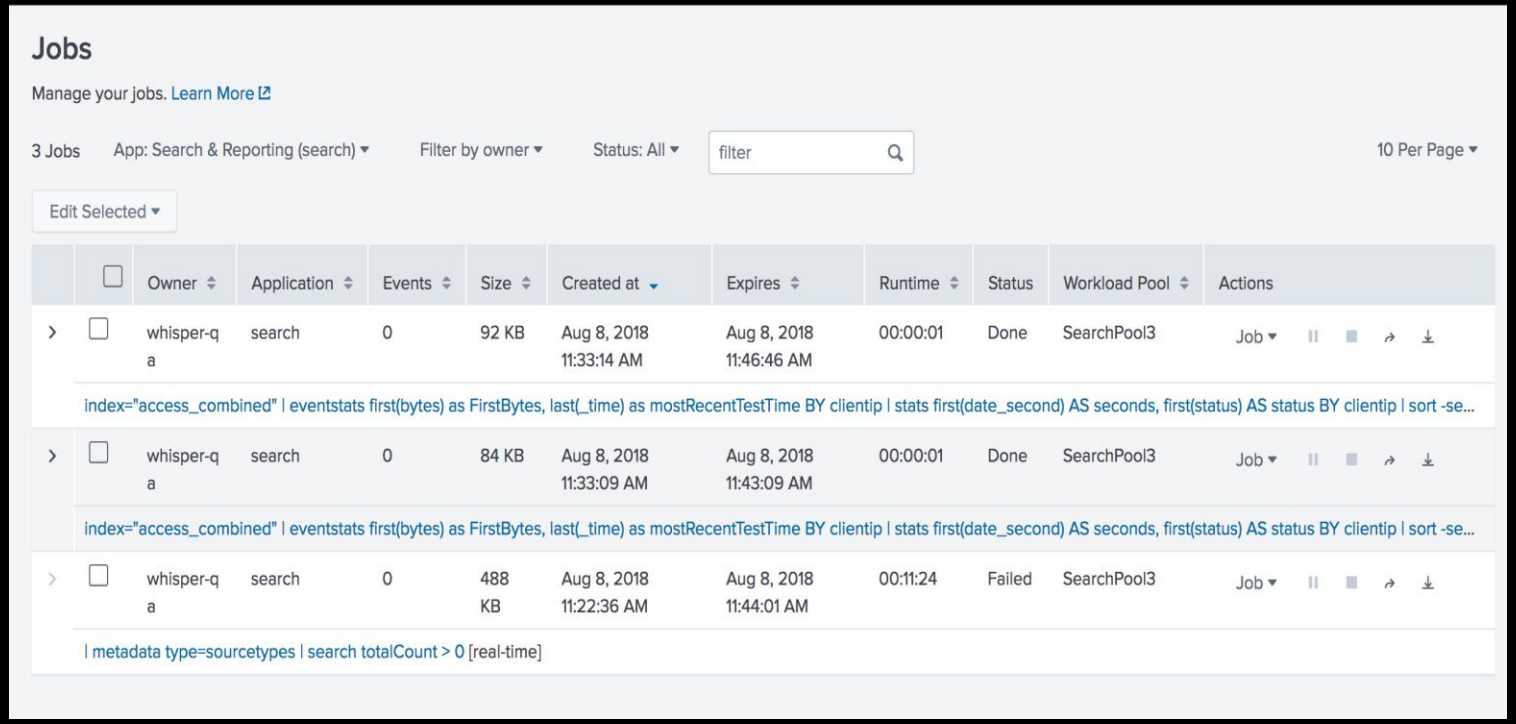
Capability	Actions
edit_workload_pools	Create/Modify Workload Pools
list_workload_pools	List Workload Pools
select_workload_pools	Select Workload Pools
edit_workload_rules	Create/Modify Workload Rules
list_workload_rules	List Workload Rules

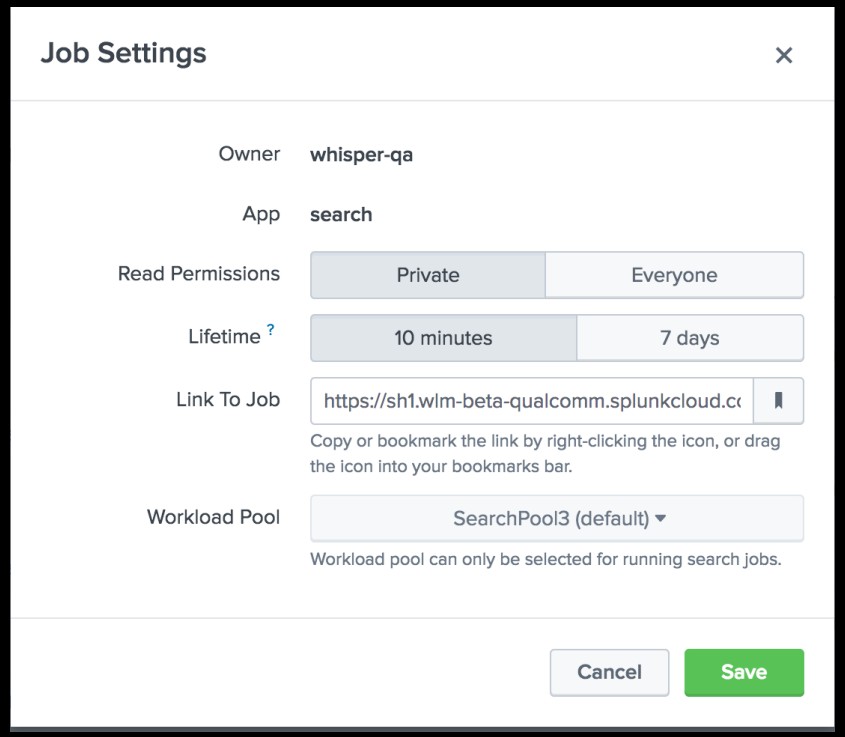


Dynamic Pool Selection: Ad-hoc Searches



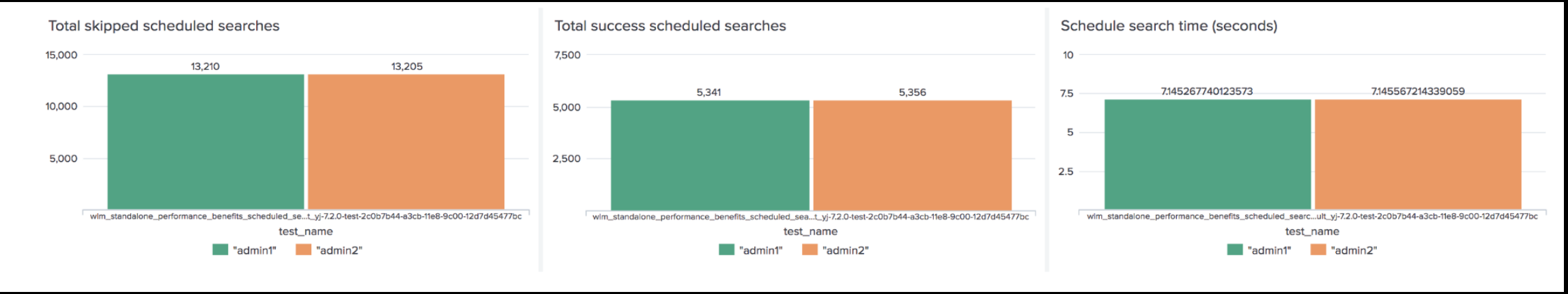
Dynamic Pool Selection: Scheduled Searches



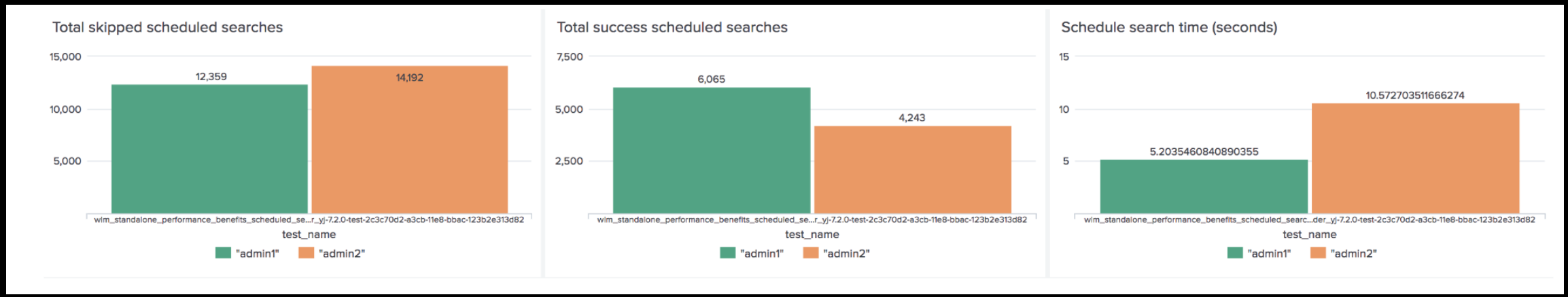


Scheduled Search Behavior across Workload Pools

- WLM Disabled



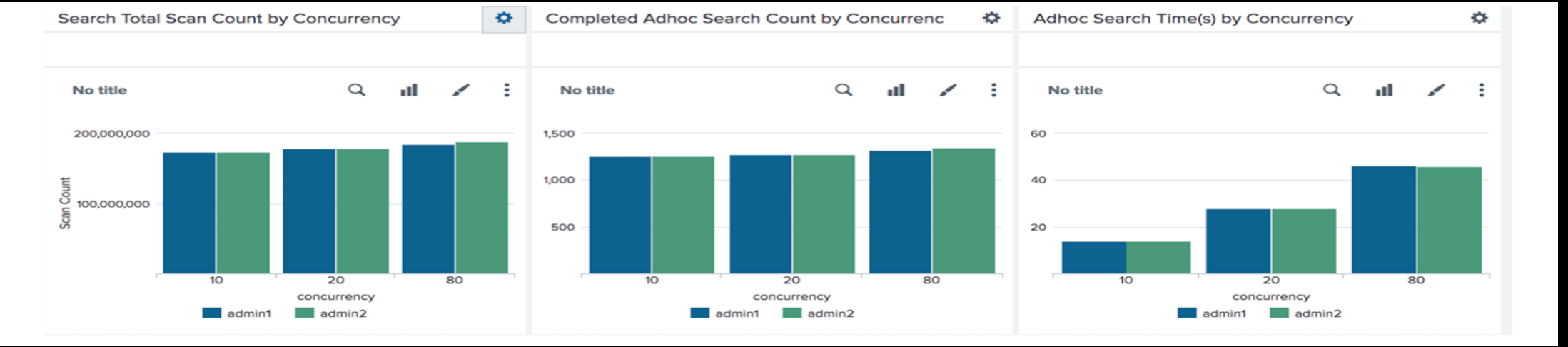
- WLM Enabled



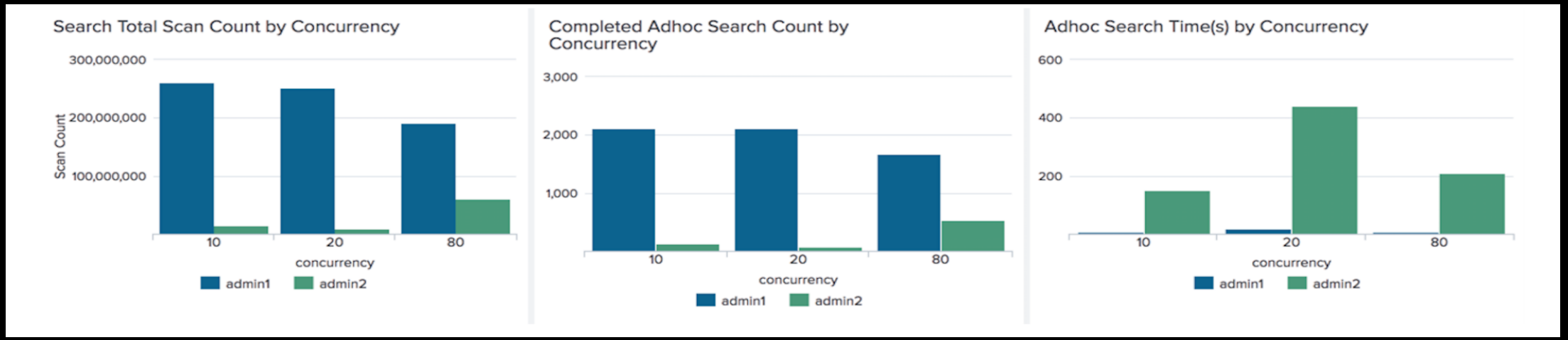
- Scheduled searches run faster in higher resource workload pool
- Fewer skipped searches in higher resource workload pool
- Higher number of successfully scheduled searches in higher resource workload pool

Ad-hoc Search Behavior across Workload Pools

- WLM Disabled



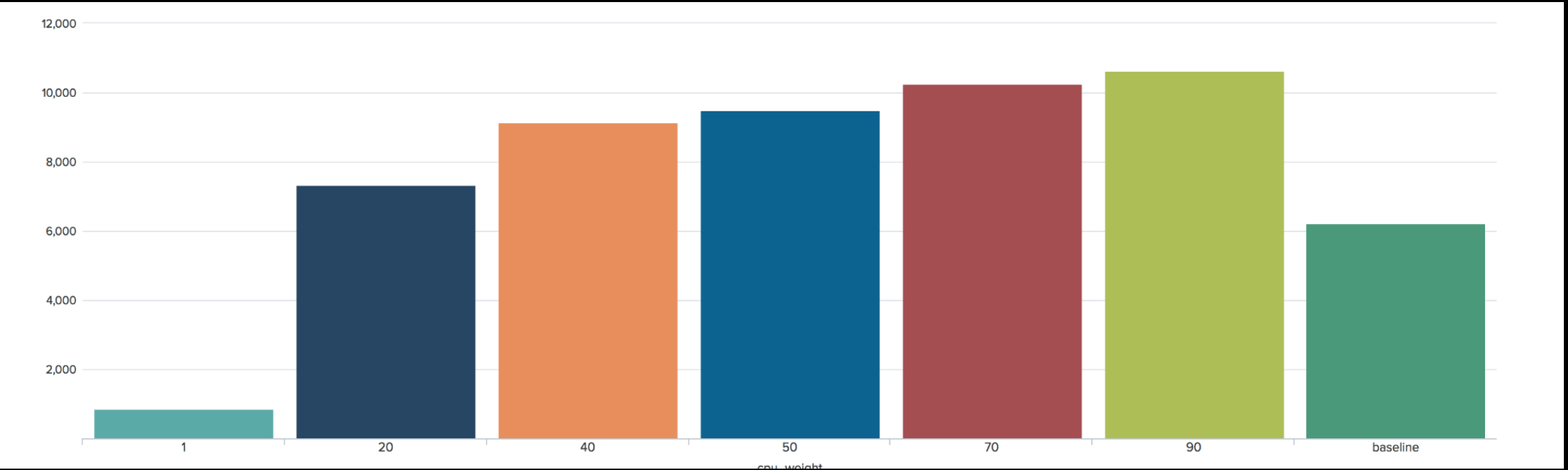
- WLM Enabled



- Ad-hoc searches run faster in higher resource workload pool
- Higher number of successfully completed ad-hoc searches by concurrency in higher resource workload pool

splunk> .conf18

Ingest Behavior across Workload Pools



- Increased ingest throughput under CPU load for higher resource workload pool
- Throughput will be affected if resources are under-provisioned for Ingest pool
- Parallel ingestion pipeline sets can help achieve better performance with sufficient resource allocation for Ingest workload pool

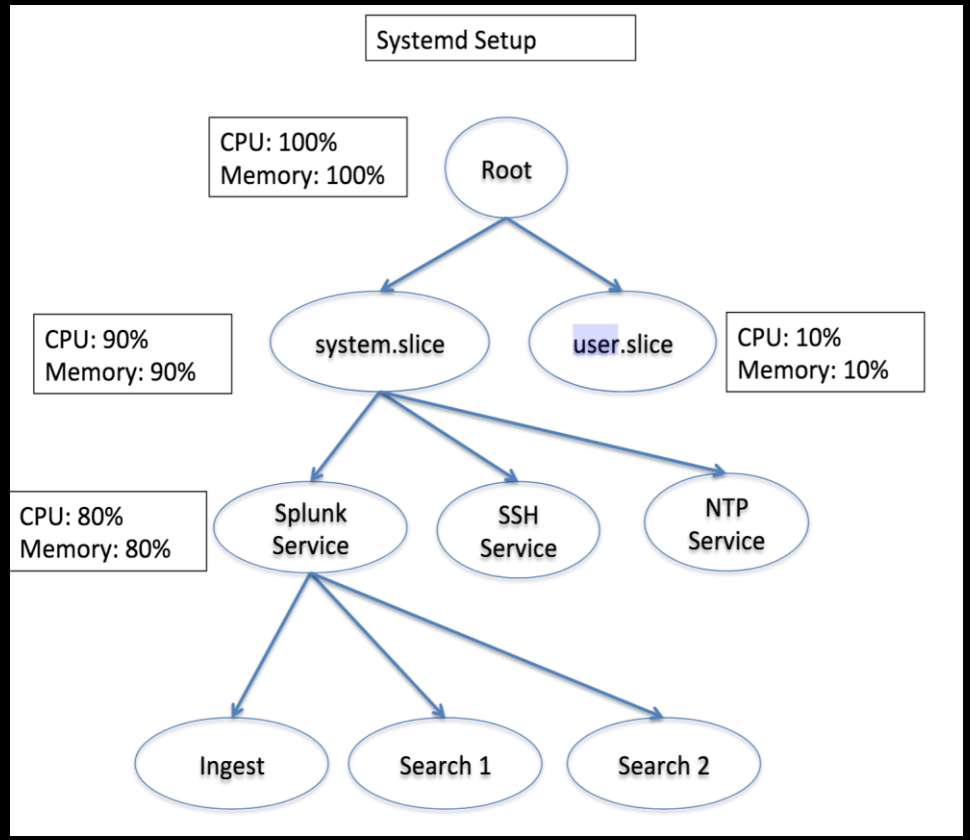
Deep Dive

Setup, Configuration and Diagnostics

splunk> .conf18

Setup

- Systemd
  - Create a systemd unit file in /etc/systemd/system/systemd.service
    - Sample unit file available in the docs
  - Run ”systemctl daemon-reload” to reload the unit file
  - Run splunk start. This starts splunkd as a systemd service.
  - Verify that splunkd is running as a systemd service
    systemctl status SPLUNK_SERVER_NAME.service



- Assign CPU for “splunk” cgroup
  - Total system CPU shares in /sys/fs/cgroup/cpu/cpu.shares
  - Set CPU in
    - /sys/fs/cgroup/cpu/system.slice to 9126
    - /sys/fs/cgroup/cpu/system.slice/splunkd.service /cpu.shares to 9126

- Assign physical memory for “splunk” cgroup
  - Total system physical memory from /proc/meminfo
  - Set Memory in
    - /sys/fs/cgroup/memory/memory.limit_in_bytes to100% of total physical memory
    - /sys/fs/cgroup/memory/system.slice/memory.limit_in_bytes tp 90% of total physical memory
    - /sys/fs/cgroup/memory/system.slice/splunkd.service/memory.limit_in_bytes to 80% of physical memory

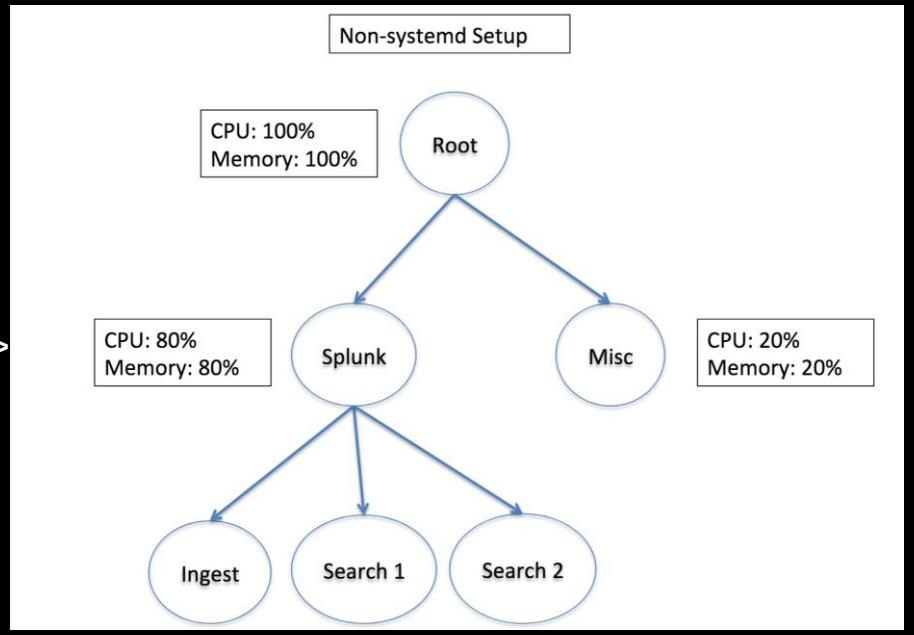
Sample Unit File

```
[Service]
Restart=always
Type=simple
ExecStart=/root/rdimri/splunk/bin/splunk _internal_launch_under_systemd --accept-license --no-prompt --answer-yes --seed-passwd changeme
Delegate=true

#Splunk defines successful exit codes other than 0 to indicate special exit scenarios which are
#used by splunk operations like rolling-restart, offline etc.
SuccessExitStatus=51 52
RestartPreventExitStatus=51
RestartForceExitStatus=52

#On some systemd installations, systemd does not create cgroups for memory and cpu controller under system.slice
#rather it runs process under root cgroups, we can force systemd to create cgroups under system.slice by specifying
#MemoryLimit and CPUShares, please look at description below.
MemoryLimit=100G
CPUShares=1024

#If you want to run splunk as non-root user uncomment the following four lines.
PermissionsStartOnly=true
User=splunk
Group=splunk
ExecStartPost=/bin/bash -c "chown -R <USER Specified above>:<GROUP of User> /sys/fs/cgroup/cpu/system.slice/%n"
ExecStartPost=/bin/bash -c "chown -R <USER Specified above>:<GROUP of User> /sys/fs/cgroup/memory/system.slice/%n"
```

splunk> .conf18

Setup

- Non-Systemd (Initd)
  - Set splunk root cgroup in workload_pools.conf
    [general]
    workload_pool_base_dir_name = splunk
  - Create cpu and memory cgroups
    - sudo mkdir /sys/fs/cgroup/cpu/<workload_pool_base_dir_name>
    - sudo mkdir /sys/fs/cgroup/memory/<workload_pool_base_dir_name>
    - sudo chown -R ${USER} /sys/fs/cgroup/cpu/<workload_pool_base_dir_name>
    - sudo chown -R ${USER} /sys/fs/cgroup/memory/<workload_pool_base_dir_name>
- Assign CPU for “splunk” cgroup
  - Total system CPU shares in /sys/fs/cgroup/cpu/cpu.shares
  - Set CPU in /sys/fs/cgroup/cpu/splunk/cpu.shares (80% of total cpu)
- Assign physical memory for “splunk” cgroup
  - Total system physical memory from /proc/meminfo
  - Set Memory in /sys/fs/cgroup/memory/splunk/memory.limit_in_bytes (80% of physical memory)



Configuration

- On a search head, perform the below steps through UI/CLI/REST
  - Create workload groups
  - Create workload rules
  - Enable workload Management
  - Check workload management status
    - splunk show workload-management-status
  - If search head is part of a search head cluster, changes are propagated to the other cluster members

- Indexer Cluster Configuration
  - Copy workload_pools.conf from search head to the CM
    - CM location: $SPLUNK_HOME/etc/master-apps/_cluster/local directory
  - Execute CM bundle push from the CM

Interaction with Existing Search Quota Settings

- Existing search quota and scheduler priority changes continue to be applied along with Splunk Workload Management.
- Workload management does not attempt to override the settings described below.

Setting Name	Configuration File	Description	Default Value
srchJobsQuota	authorize.conf	Max number of concurrent historical searches by role	3
cumulativeSrchJobsQuota	authorize.conf	Max number of concurrent historical searches for all members of role	N/A
base_max_searches	limits.conf	Constant to add to max no of searches as a factor of CPUs	6
max_searches_per_cpu	limits.conf	Max number of concurrent historical searches per CPU	1
schedule_priority	savedsearches.conf	Raises scheduling priority of a search. Can have value: default, higher, highest.	default

Key Takeaways

1. First level bullets should be sentence case, 28pt
2. First level bullets should be sentence case, 28pt
3. First level bullets should be sentence case, 28pt

Q&A

splunk> .conf18

Thank You

Don't forget to rate this session in the .conf18 mobile app