Splunk 101/201 Hands-on Workshops

splunk>

Forward-looking statements

splunk>®

Workshop Logistics

See Zoom chat for Splunk workshop instance details

Workshop flow:

- **LEARN:** New capability and outcomes on each slide
- **PRACTICE:** Apply techniques in a demo environment after each slide
- **WORK:** Apply techniques in your environment with your Splunk team after the session

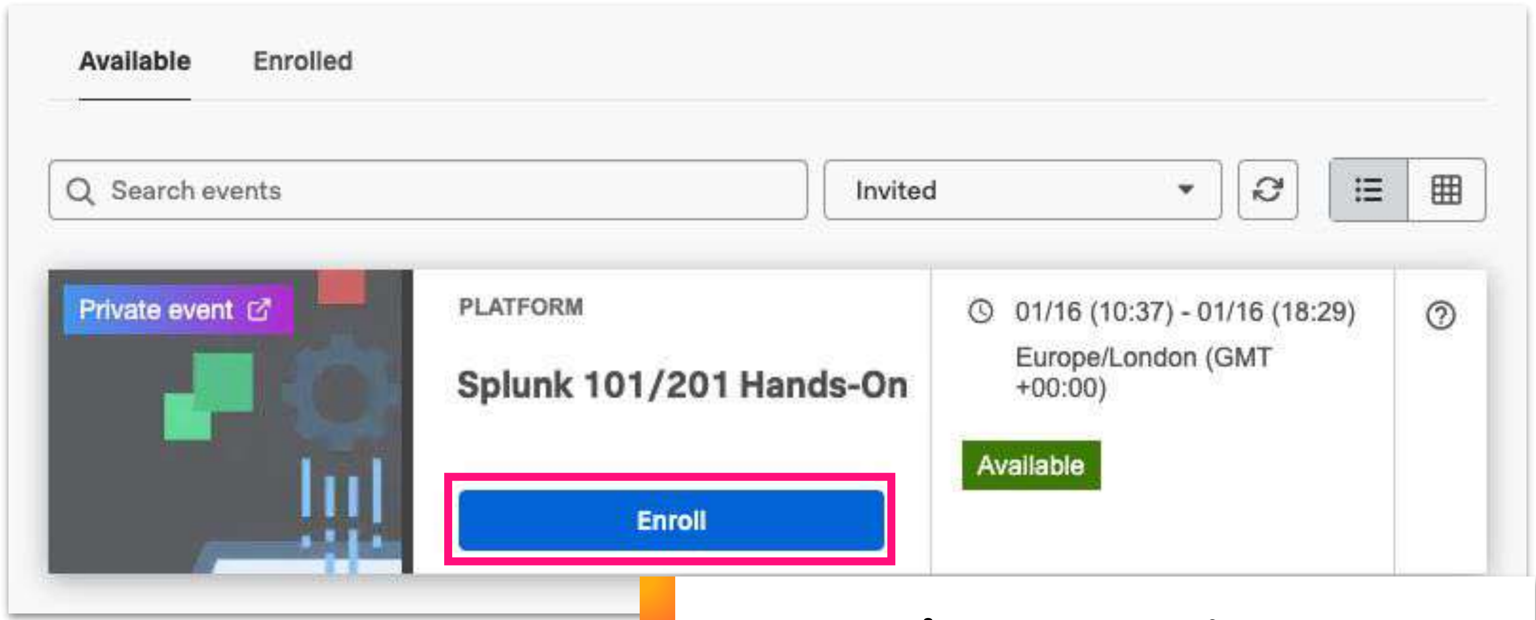
Ask questions! What outcomes are you trying to achieve?

Enroll in Today's Workshop

Tasks

1. Get a splunk.com account if you don't have one yet: https://splk.it/SignUp
2. Enroll in the Splunk Show workshop event: https://show.splunk.com/event/<eventID>
3. Download a copy of today's slide deck: https://splk.it/101-201-Attendee

Goal



Enroll in today's event

Introduction

splunk>

Splunk 101 Session Overview

Session Goals:

- Enable your teams to search, investigate, analyze & report on data
- Drive more value and better outcomes from machine data
- Increase teams' productivity and efficiency in resolving issues

Splunk 101 Agenda (for users):

- Search UI
- Anatomy of a Search
- Search Best Practices
- Open chevron, event details, actions, etc.
- Customized time spans
- search commands, Booleans, save as dashboard
- stats, timechart, eval & where functions
- rename, dedup, fields/table, head/rare/top functions
- Common use cases in Security, IT, and Observability

Splunk 201 Session Overview

Session Goals:

- Enable your teams to search, investigate, analyze & report on data
- Drive more value and better outcomes from machine data
- Increase teams' productivity and efficiency in resolving issues

Splunk 201 Agenda (for power users):

- Transactional analysis (stats list/values, transaction, histograms)
- Advanced statistics (stats stdev/variance, eventstats & streamstats)
- Advanced Dashboarding (form inputs, tokens, post-process search,
- Dashboard Studio)
- Schema-on-read and Field Extraction (rex, Interactive Field Extractor,
- eventtypes/tags/macros, Common Information Model)

Splunk 301 for Security and IT/Observability

We have lots of 301-level workshops available!

Security 301 Workshops include:

- Security Operations Suite Hands-On Workshop
- Enterprise Security Hands-On Workshop

IT/Observability 301 Workshops include:

- Business Service Insights Workshop
- ITSI Event Analytics Workshop
- Splunk4Ninjas - IT Service Intelligence (ITSI)

For more options, reach out to your Splunk account team and we can facilitate based on your use cases

Additional Resources

Splunk Docs (with detailed reference for all commands):
https://docs.splunk.com/

FREE Splunk Fundamentalstraining:
https://www.splunk.com/en_us/training/free-courses/overview.html

Splunk Lantern (best practices & use cases):
https://lantern.splunk.com

Splunk Community (answers to common & uncommon questions):
https://community.splunk.com/

.conf Online (incl. past keynotes & customer presentations):
https://conf.splunk.com/

Splunk Quick Reference Guide:
https://www.splunk.com/pdfs/solution-guides/splunk-quick-reference-guide.pdf

Splunk Use Cases & Customer Examples

IT Operations

ING

Predict service-level degradation before it occurs

Application Observability

Adobe

Improve app performance and reliability using app logs and infrastructure

Security & Compliance

Seal of the City and County of San Francisco

Speed up security investigations and reduce the impact of insider threats

Business Analytics

Domino's

Drive more orders with marketing campaigns across website and mobile

Internet of Things

BOSCH

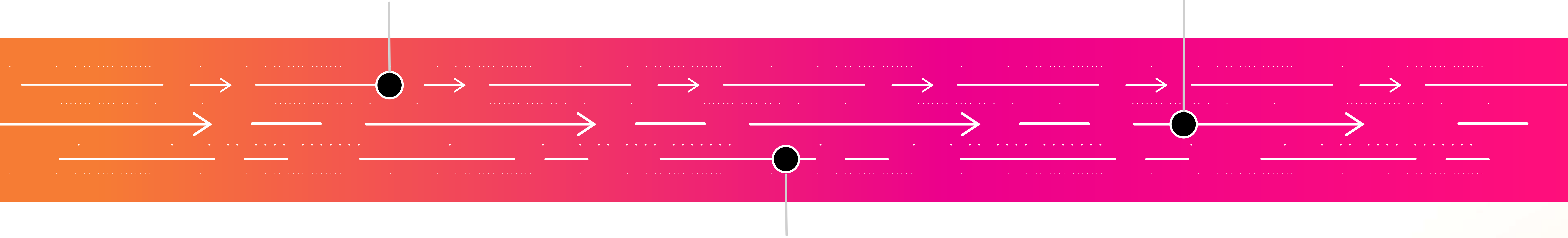
Monitor and resolve problems from 10,000s of sensors in real time

Our Investigative Approach

Adaptable | Real-Time | Fast To Value | Massive Scale

Send
unstructured data from all systems, devices, and people

React
quickly to changing circumstances by asking questions immediately

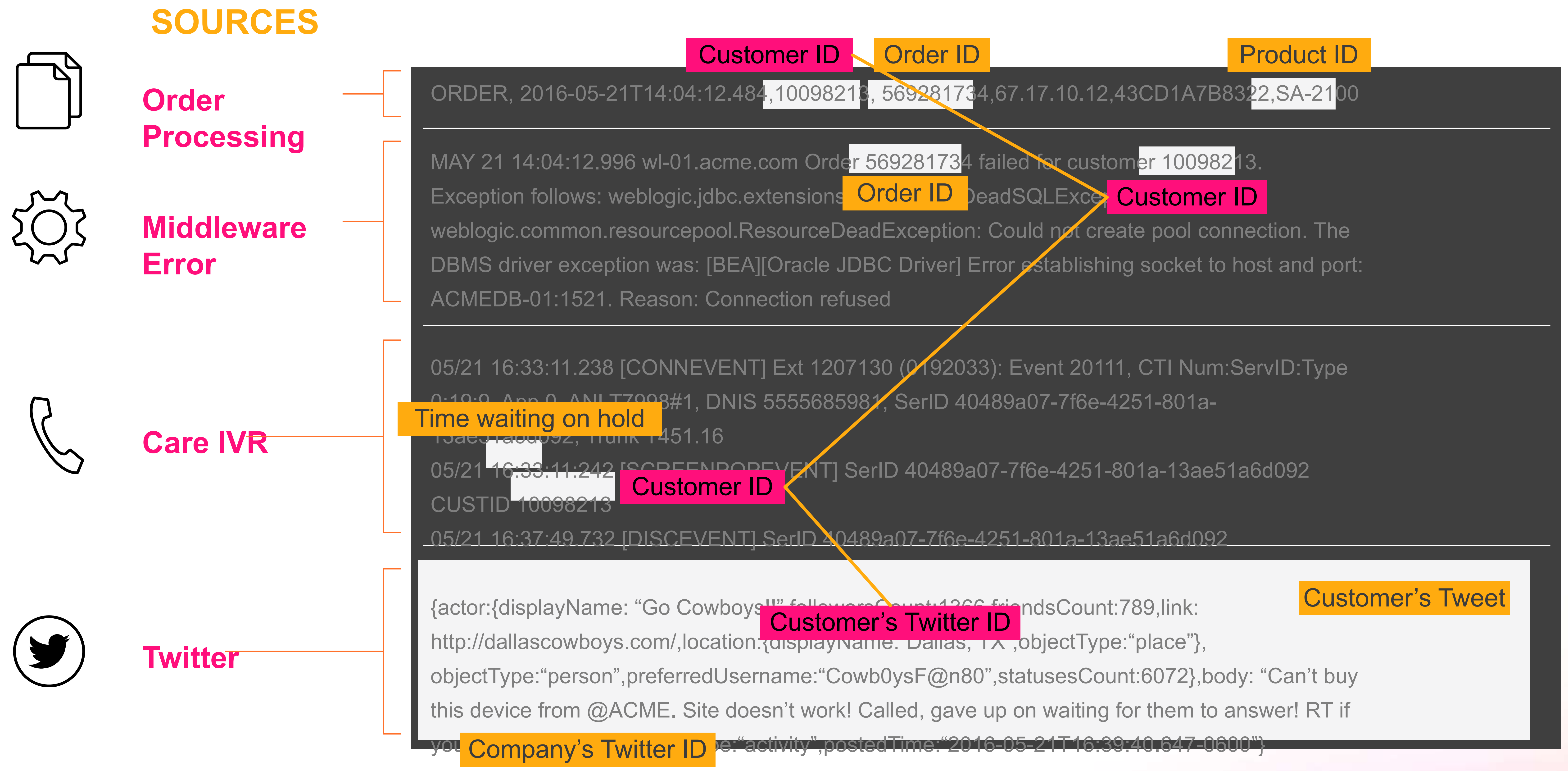


Don’t Structure
your data until you are ready

Splunk 101

splunk>

Reduce impact by observing end-to-end transactions



```
ORDER, 2016-05-21T14:04:12.484,10098213, 569281734,67.17.10.12,43CD1A7B8322,SA-2100
```

```
MAY 21 14:04:12.996 wl-01.acme.com Order 569281734 failed for customer 10098213.
Exception follows: weblogic.jdbc.extensions DeadSQLExce
weblogic.common.resourcepool.ResourceDeadException: Could not create pool connection. The
DBMS driver exception was: [BEA][Oracle JDBC Driver] Error establishing socket to host and port:
ACMEDB-01:1521. Reason: Connection refused
```

```
05/21 16:33:11.238 [CONNEVENT] Ext 1207130 (0192033): Event 20111, CTI Num:ServID:Type
0:19:0, App 0, ANI T7998#1, DNIS 5555685981, SerID 40489a07-7f6e-4251-801a-
13ae51a6d092, Trunk T451.16
05/21 16:33:11.242 [SCREENPOPEVENT] SerID 40489a07-7f6e-4251-801a-13ae51a6d092
CUSTID 10098213
05/21 16:37:49.732 [DISCEVENT] SerID 40489a07-7f6e-4251-801a-13ae51a6d092
```

```
{actor:{displayName: "Go Cowboys!!",followersCount:1366,friendsCount:789,link:
http://dallascowboys.com/,location:{displayName: "Dallas, TX",objectType:"place"},
objectType:"person",preferredUsername:"Cowb0ysF@n80",statusesCount:6072},body: "Can't buy
this device from @ACME. Site doesn't work! Called, gave up on waiting for them to answer! RT if
you pe:"activity",postedTime:"2016-05-21T16:39:40.647-0600"}
```

Search UI

Search & investigate logs for errors:

- `sourcetype=* error`

Quick reports:

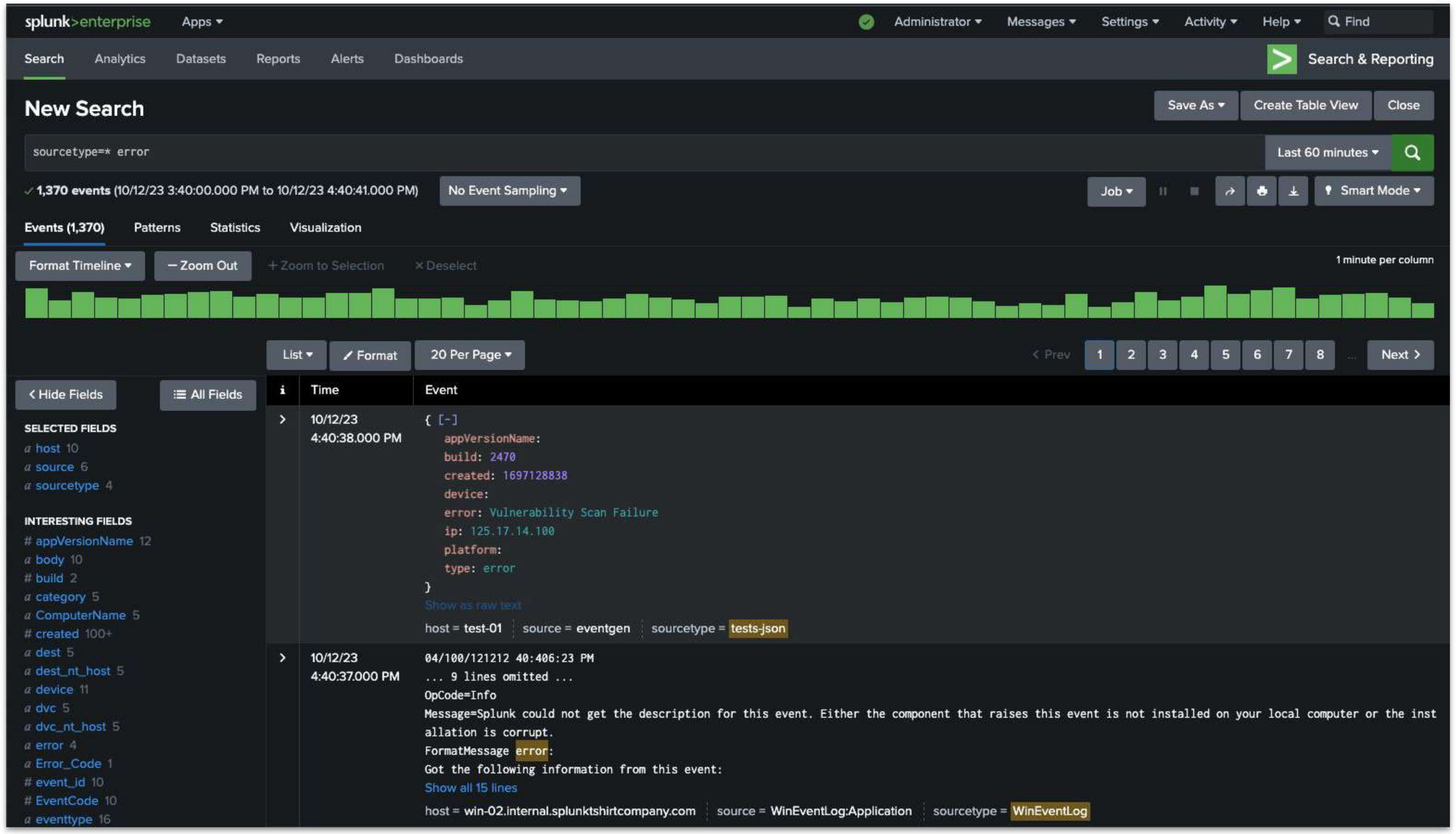
- Top values (by time), rare values
- Events with this field
- Average/Maximum/Minimum over time

Customize time spans

- Relative, Real-time, Date & Time Range

Outcomes & value:

- Find needle-in-haystack problems or threats
- Reduce Mean Time To Resolve (MTTR) issues



```
sourcetype=* error
```

✓ 1,370 events (10/12/23 3:40:00.000 PM to 10/12/23 4:40:41.000 PM)

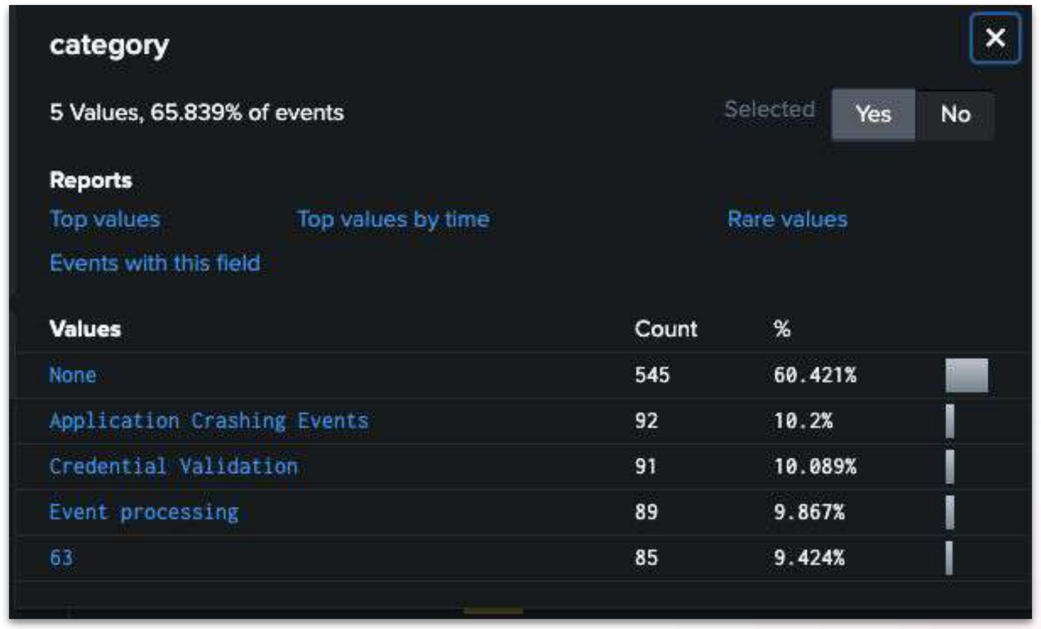
SELECTED FIELDS

a host	10
a source	6
a sourcetype	4

INTERESTING FIELDS

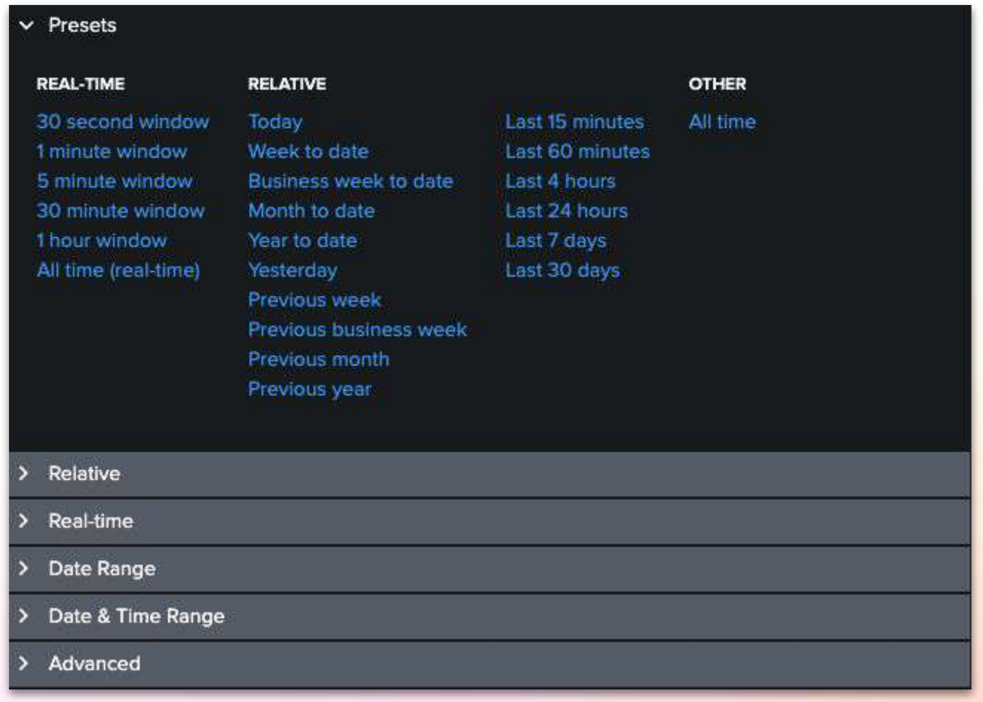
# appVersionName	12
a body	10
# build	2
a category	5
a ComputerName	5
# created	100+
a dest	5
a dest_nt_host	5
a device	11
a dvc	5
a dvc_nt_host	5
a error	4
a Error_Code	1
# event_id	10
# EventCode	10
a eventtype	16

i	Time	Event
>	10/12/23 4:40:38.000 PM	`{ [-] appVersionName: build: 2470 created: 1697128838 device: error: Vulnerability Scan Failure ip: 125.17.14.100 platform: type: error }` Show as raw text host = test-01 source = eventgen sourcetype = tests-json
>	10/12/23 4:40:37.000 PM	`04/100/121212 40:406:23 PM ... 9 lines omitted ... OpCode=Info Message=Splunk could not get the description for this event. Either the component that raises this event is not installed on your local computer or the installation is corrupt. FormatMessage error: Got the following information from this event:` Show all 15 lines host = win-02.internal.splunktshirtcompany.com source = WinEventLog:Application sourcetype = WinEventLog



5 Values, 65.839% of events

Values	Count	%
None	545	60.421%
Application Crashing Events	92	10.2%
Credential Validation	91	10.089%
Event processing	89	9.867%
63	85	9.424%



REAL-TIME	RELATIVE		OTHER
30 second window	Today	Last 15 minutes	All time
1 minute window	Week to date	Last 60 minutes	
5 minute window	Business week to date	Last 4 hours	
30 minute window	Month to date	Last 24 hours	
1 hour window	Year to date	Last 7 days	
All time (real-time)	Yesterday	Last 30 days	
	Previous week		
	Previous business week		
	Previous month		
	Previous year		

Search UI (cont.)

Open chevron:

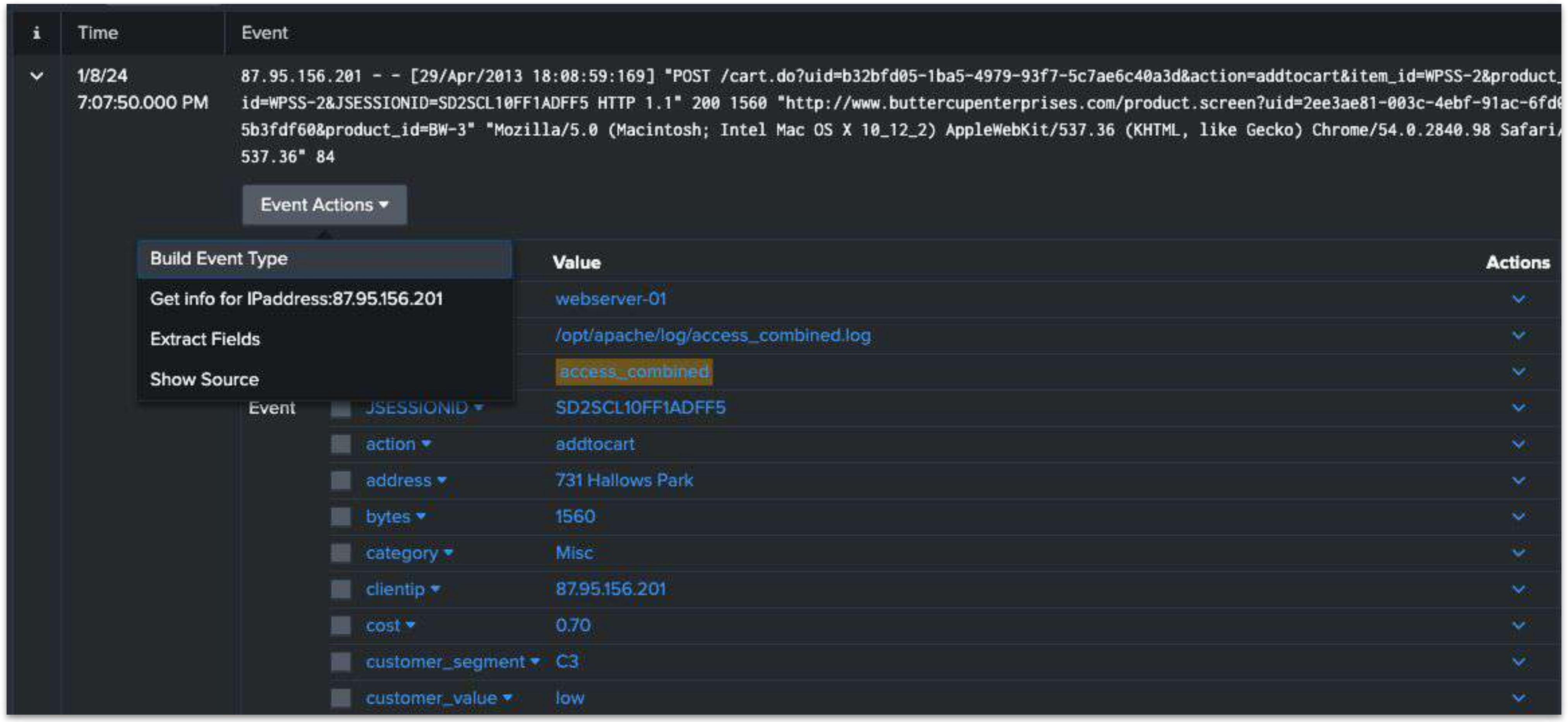
- Get event details
- Add new tags
- Add/exclude or new search from terms
- Trigger event actions & automation (Work with power users & Splunk admin team)

Search nearby events:

- Click timestamp
- Events before or after
- Modify search within new time range

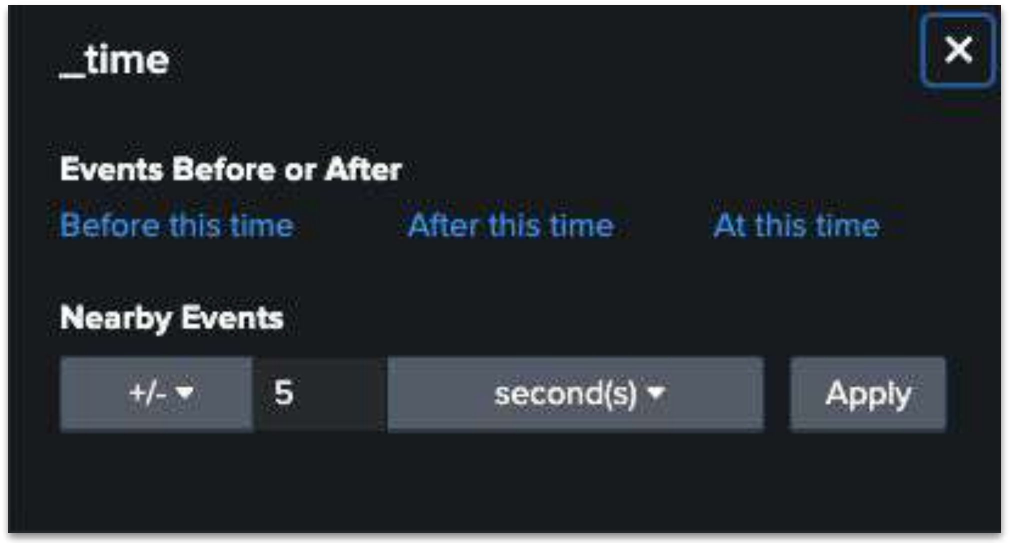
Outcomes & value:

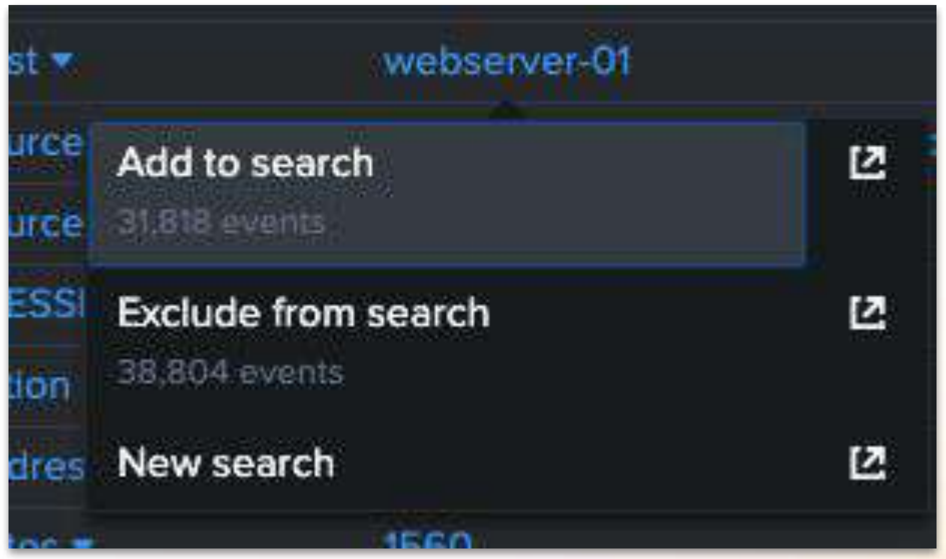
- Reduce MTTR by surfacing related events via metadata, keywords, and time



i	Time	Event
⌄	1/8/24 7:07:50.000 PM	87.95.156.201 - - [29/Apr/2013 18:08:59:169] "POST /cart.do?uid=b32bfd05-1ba5-4979-93f7-5c7ae6c40a3d&action=addtocart&item_id=WPSS-2&product_id=WPSS-2&JSESSIONID=SD2SCL10FF1ADFF5 HTTP 1.1" 200 1560 "http://www.buttercupenterprises.com/product.screen?uid=2ee3ae81-003c-4ebf-91ac-6fd 5b3fdf60&product_id=BW-3" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_12_2) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/54.0.2840.98 Safari/ 537.36" 84

Type	Field	Value	Actions
		webserver-01	⌄
		/opt/apache/log/access_combined.log	⌄
		access_combined	⌄
Event	JSESSIONID ▾	SD2SCL10FF1ADFF5	⌄
	action ▾	addtocart	⌄
	address ▾	731 Hallows Park	⌄
	bytes ▾	1560	⌄
	category ▾	Misc	⌄
	clientip ▾	87.95.156.201	⌄
	cost ▾	0.70	⌄
	customer_segment ▾	C3	⌄
	customer_value ▾	low	⌄





https://docs.splunk.com/Documentation/Splunk/latest/Search/Usetimeaccelerators

Search with Booleans

Combine events with Booleans (OR, AND, NOT)

Search combined app log + web log data:

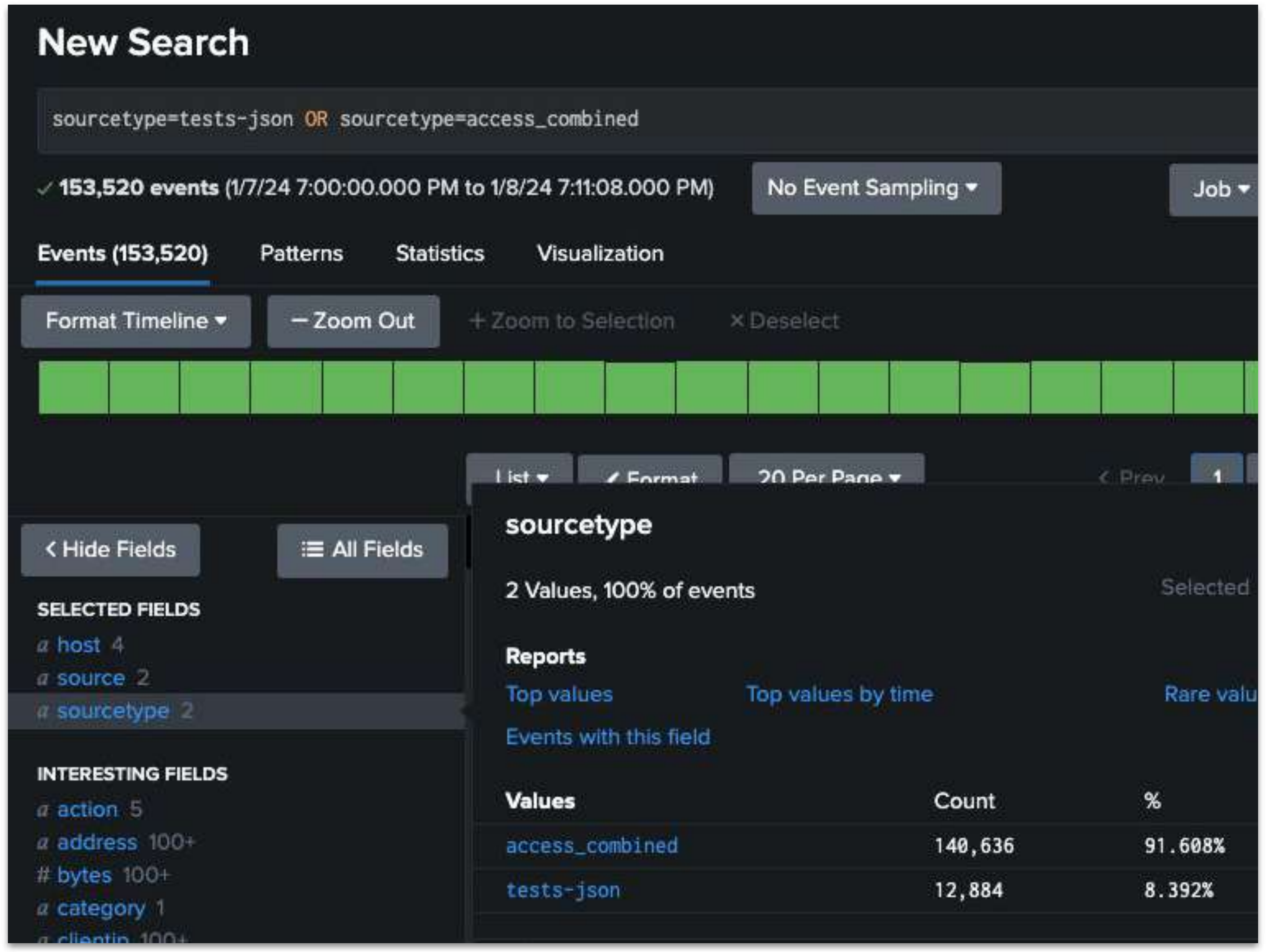
- `sourcetype=tests-json OR sourcetype=access_combined`

Search app log data for host X AND platform Y:

- `sourcetype=tests-json host=test-01 platform=iOS`

Search app log data for host X AND NOT platform Y:

- `sourcetype=tests-json host=test-01 NOT platform=iOS`



New Search

```
sourcetype=tests-json OR sourcetype=access_combined
```

✓ **153,520 events** (1/7/24 7:00:00.000 PM to 1/8/24 7:11:08.000 PM)

SELECTED FIELDS
a host 4
a source 2
a sourcetype 2

INTERESTING FIELDS
a action 5
a address 100+
# bytes 100+
a category 1
a clientip 100+

sourcetype

2 Values, 100% of events

Selected

Reports

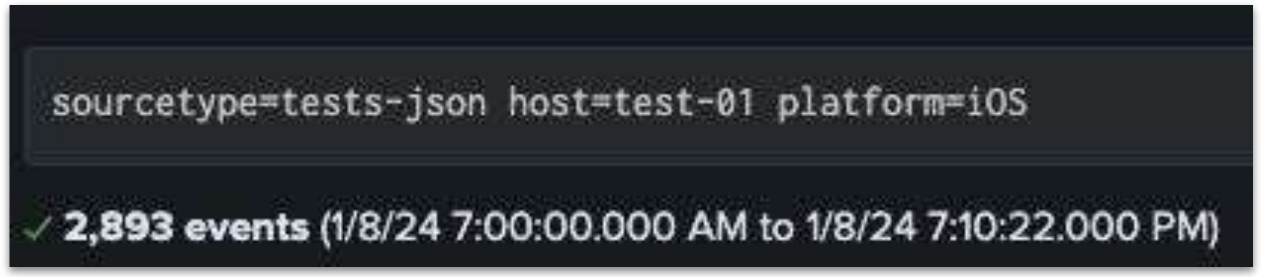
Top values · Top values by time · Rare valu

Events with this field

Values	Count	%
access_combined	140,636	91.608%
tests-json	12,884	8.392%

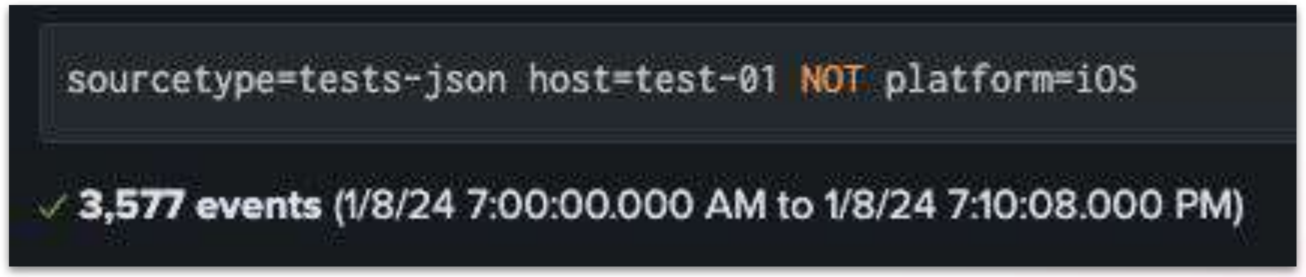
Outcomes & value:

- Detect and resolve issues by narrowing in on exact events (needles in haystacks)



```
sourcetype=tests-json host=test-01 platform=iOS
```

✓ **2,893 events** (1/8/24 7:00:00.000 AM to 1/8/24 7:10:22.000 PM)



```
sourcetype=tests-json host=test-01 NOT platform=iOS
```

✓ **3,577 events** (1/8/24 7:00:00.000 AM to 1/8/24 7:10:08.000 PM)

Anatomy of a Search

Search Query Structure

```
index=waldo error OR fail*| eval loc=long+lat+alt | geoip loc
```

retrieve events

filter/transform/operate/map

Search Best Practices

Search for just what you need:

- Small time ranges (e.g., 15 minutes, not all time)
- Exact indexes and sourcetypes:
  - `index=oidemo (sourcetype=access_combined OR sourcetype=tests-json)`
- Use eventtypes and tags for consistent and repeatable searches:
  - `tag=network`
  - `eventtype=nix-all-logs`

Reduce load on system with optimal exact techniques instead of easy ones:

- More efficient and optimal with “stats list” or “stats value” commands:
  - `index=oidemo sourcetype=access_combined | stats list(action) by JSESSIONID`
- Less efficient but easy with “transaction” command:
  - `index=oidemo sourcetype=access_combined | transaction JSESSIONID`

Outcomes & value:

- Reduce MTTR by pulling back exact data and running faster searches
- Reduce impact to other users and deliver more efficient Splunk service

Search & stats

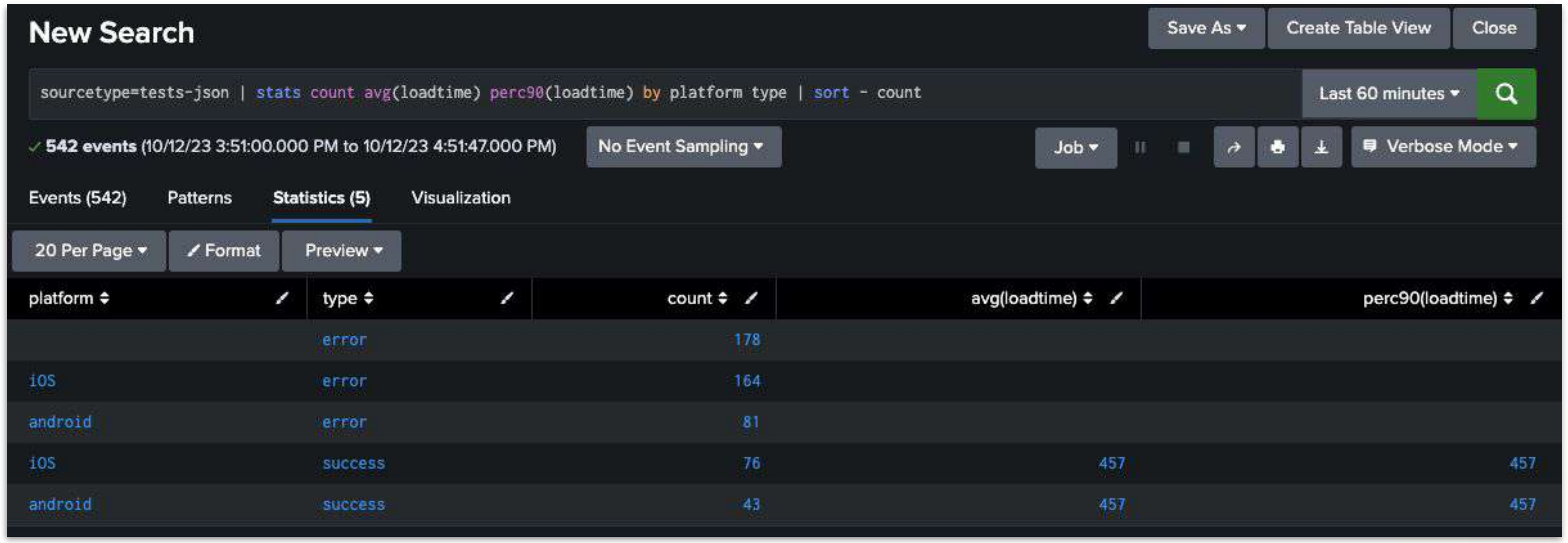
Use stats command to turn data into aggregate tables

Example: Track errors vs successes and load times by platform:

- `sourcetype=tests-json | stats count avg(loadtime) perc90(loadtime) by platform type | sort - count`

Statistical functions:

- `c` / `count` / `sum` / `distinct_count` / `dc`
- `min` / `max`
- `avg` / `stdev`
- `median` / `perc90 perc<int1-100>`
- `list` / `values`
- `earliest` / `latest`



```
sourcetype=tests-json | stats count avg(loadtime) perc90(loadtime) by platform type | sort - count
```

✓ 542 events (10/12/23 3:51:00.000 PM to 10/12/23 4:51:47.000 PM)

platform	type	count	avg(loadtime)	perc90(loadtime)
	error	178		
iOS	error	164		
android	error	81		
iOS	success	76	457	457
android	success	43	457	457

Outcomes & value:

- Group data to find patterns to detect future problems, report on current situation & context

https://docs.splunk.com/Documentation/Splunk/latest/SearchReference/Stats

Search & timechart

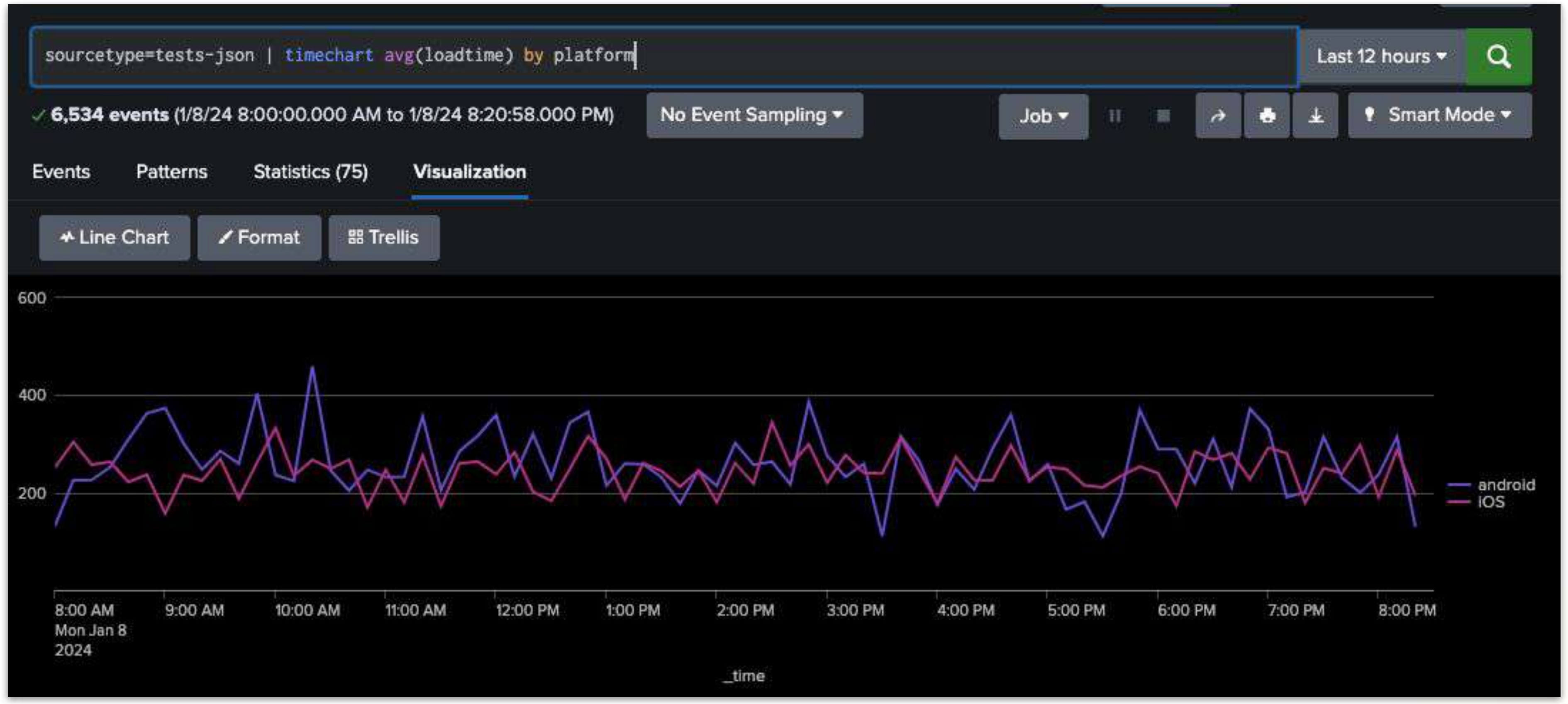
Use timechart command to turn data into timecharts

Example: Chart median loadtimes across platforms:

- `sourcetype=tests-json | timechart avg(loadtime) by platform`

Statistical functions:

- count / sum
- min / max
- avg / stdev
- median / perc90
- earliest / latest
- distinct_count / dc / estdc



```
sourcetype=tests-json | timechart avg(loadtime) by platform
```

✓ 6,534 events (1/8/24 8:00:00.000 AM to 1/8/24 8:20:58.000 PM)

Outcomes & value:

- Report and analyze patterns and trends, communicate anomalies to management

https://docs.splunk.com/Documentation/Splunk/latest/SearchReference/Timechart

Save As Report, Private Dashboard or Alert

Save any search as report, private dashboard, or alert

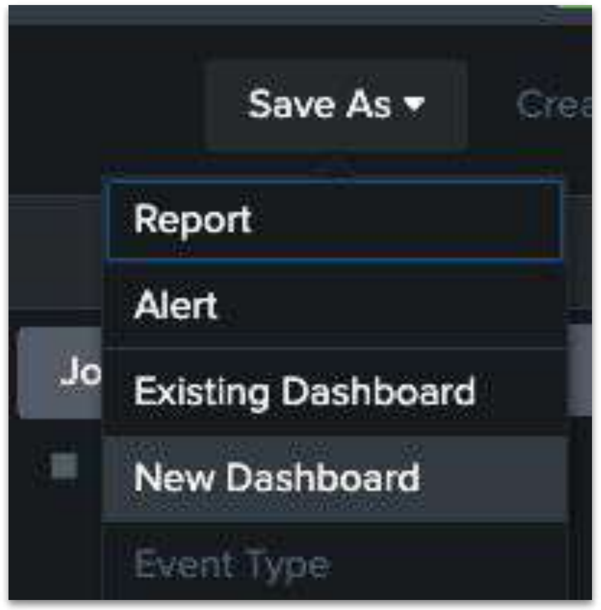
- Click ”Save As” then “Report” or “Dashboard Panel” or “Alert”

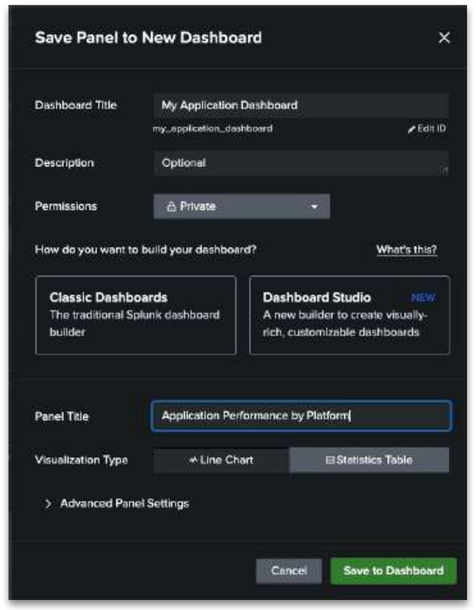
Edit visualizations, add form inputs, adjust to your needs

Work with power users & Splunk admin team to share & publish content (dashboards, alerts)

Outcomes & value:

- Improve productivity by collaborating with team members using dashboards & reports
- Improve MTTD using alerts




Save Panel to New Dashboard

Dashboard Title: My Application Dashboard
my_application_dashboard
Edit ID

Description: Optional

Permissions: Private

How do you want to build your dashboard? What's this?

Classic Dashboards The traditional Splunk dashboard builder

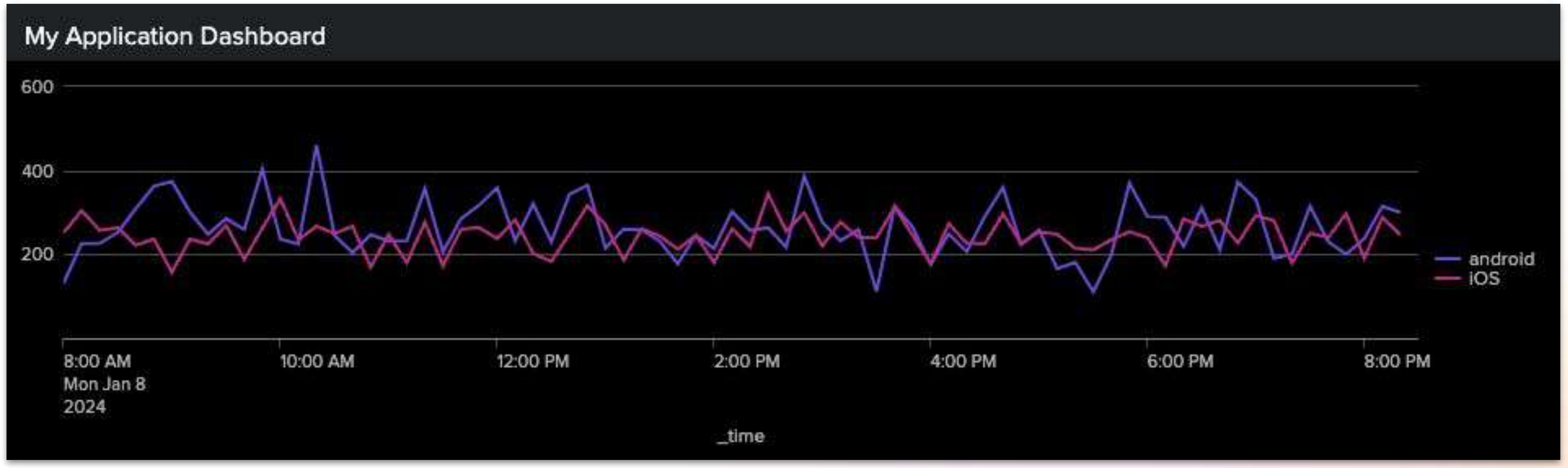
Dashboard Studio NEW A new builder to create visually-rich, customizable dashboards

Panel Title: Application Performance by Platform

Visualization Type: Line Chart | Statistics Table

Advanced Panel Settings

Cancel | Save to Dashboard

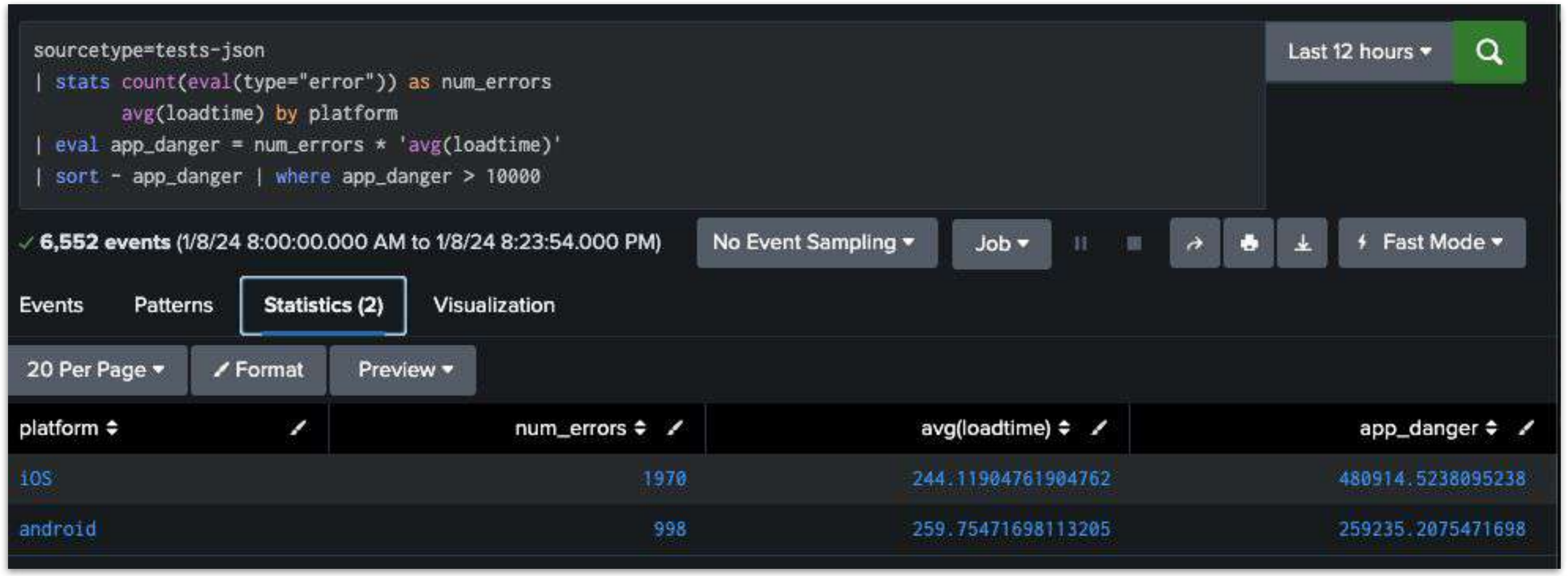


eval & where Commands

Use eval command to calculate expressions and create new fields
Use where command to filter based on Booleans

Example: Find apps with high errors or load times:

-
  ```
  sourcetype=tests-json
  | stats count(eval(type="error")) as num_errors
          avg(loadtime) by platform
  | eval app_danger = num_errors * 'avg(loadtime)'
  | sort - app_danger | where app_danger > 10000
  ```



```
sourcetype=tests-json
| stats count(eval(type="error")) as num_errors
        avg(loadtime) by platform
| eval app_danger = num_errors * 'avg(loadtime)'
| sort - app_danger | where app_danger > 10000
```

✓ 6,552 events (1/8/24 8:00:00.000 AM to 1/8/24 8:23:54.000 PM)

platform	num_errors	avg(loadtime)	app_danger
iOS	1970	244.11904761904762	480914.5238095238
android	998	259.75471698113205	259235.2075471698

eval / where functions:

- +, -, *, /
- 'fieldnames' / "strings"
- if / case / coalesce
- isnull / isnotnull

Outcomes & value:

- Improve collaboration and detections with common field aliases and new combinations of fields

Eval command: https://docs.splunk.com/Documentation/Splunk/latest/SearchReference/Eval
Where command: https://docs.splunk.com/Documentation/Splunk/latest/SearchReference/Where

More Common Commands

Rename fields with rename:

- `sourcetype=tests-json | rename error as Error_Description`

Deduplicate repeated events with dedup:

- `sourcetype=tests-json | dedup ip`

Show only key fields in events or table view with fields/table:

- `sourcetype=tests-json | fields _time platform type error`
- `sourcetype=tests-json | table _time platform type error`

Get most recent/rarest/most common events with head/rare/top:

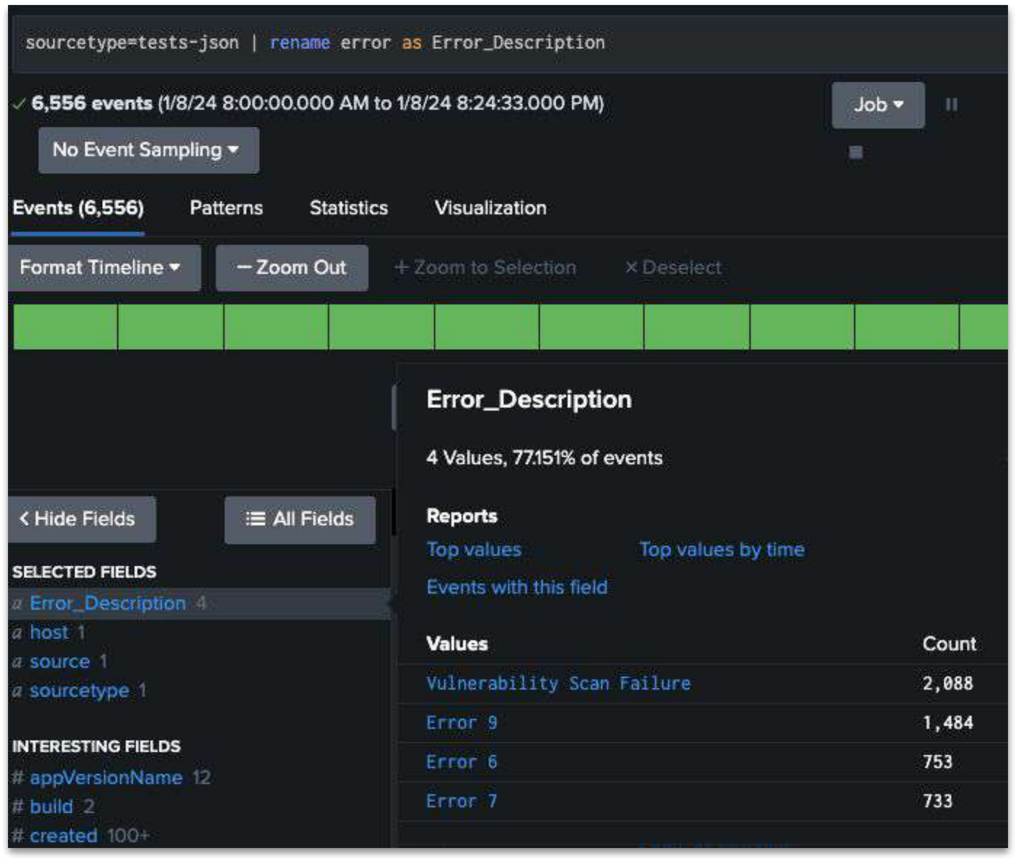
- `sourcetype=tests-json | rare platform type`

Outcomes & value:

- Detect and report on important information in your data

Plus 100+ more commands:

- https://docs.splunk.com/Documentation/Splunk/latest/SearchReference/ListOfSearchCommands



```
sourcetype=tests-json | rename error as Error_Description
```

✓ 6,556 events (1/8/24 8:00:00.000 AM to 1/8/24 8:24:33.000 PM)

SELECTED FIELDS
a Error_Description 4
a host 1
a source 1
a sourcetype 1

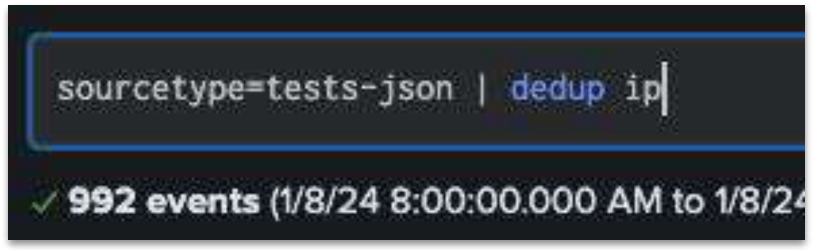
INTERESTING FIELDS
# appVersionName 12
# build 2
# created 100+

Error_Description

4 Values, 77.151% of events

Reports
Top values   Top values by time
Events with this field

Values	Count
Vulnerability Scan Failure	2,088
Error 9	1,484
Error 6	753
Error 7	733

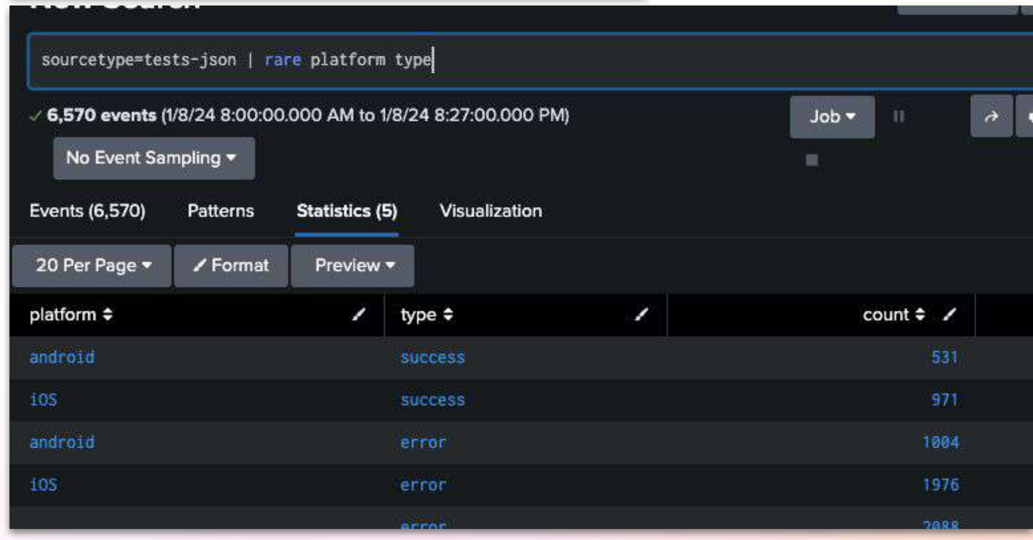


```
sourcetype=tests-json | dedup ip
```

✓ 992 events (1/8/24 8:00:00.000 AM to 1/8/24

```
sourcetype=tests-json
```

✓ 6,568 events (1/8/24 8:00:00.000



```
sourcetype=tests-json | rare platform type
```

✓ 6,570 events (1/8/24 8:00:00.000 AM to 1/8/24 8:27:00.000 PM)

platform	type	count
android	success	531
iOS	success	971
android	error	1004
iOS	error	1976
	error	2088

Common Use Cases in Security, IT & Observability

splunk>

Splunkbase

Splunkbase has 3100+ apps

Enable new use cases and extend your teams' Splunk capabilities:

https://splunkbase.splunk.com

Use cases include:

- IT Ops, Security, Observability, Business Analytics, IoT & Industrial Data
- Financial Services, Retail, Telecom, Healthcare, Energy

Technology vendors include:

- Cisco, Dell EMC, Amazon Web Services, Palo Alto Networks, Microsoft, Google

Please work with Splunk Admin team to ensure that you can install

Splunk Security Essentials

Reduce time to identify and resolve threats with out-of-the-box searches

Pre-built detections and data recommendations

- 1300+ pre-built detections from the Splunk Threat Research Team and data recommendations to stay ahead of existing and emerging threats

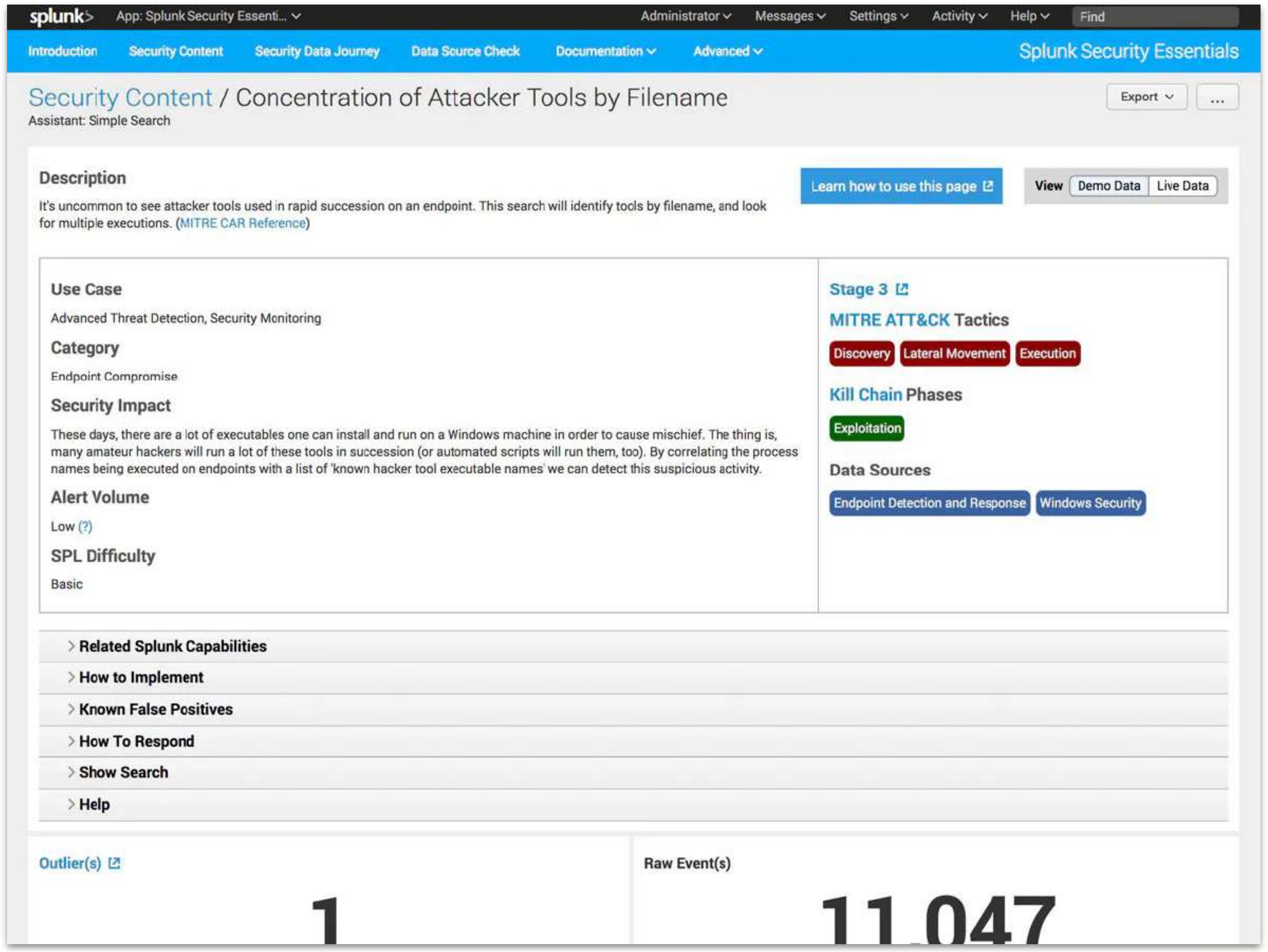
Operationalize security with industry frameworks

- Measure coverage and identify gaps in your defenses
- Maps your environment to frameworks like MITRE ATT&CK and cyber kill chain

Outcomes & value:

- Improved security detections
- More efficient Security Operations teams

Get Security Essentials for free:
https://splunkbase.splunk.com/app/3435



Security Content / Concentration of Attacker Tools by Filename

Assistant: Simple Search

Description

It's uncommon to see attacker tools used in rapid succession on an endpoint. This search will identify tools by filename, and look for multiple executions. (MITRE CAR Reference)

Use Case

Advanced Threat Detection, Security Monitoring

Category

Endpoint Compromise

Security Impact

These days, there are a lot of executables one can install and run on a Windows machine in order to cause mischief. The thing is, many amateur hackers will run a lot of these tools in succession (or automated scripts will run them, too). By correlating the process names being executed on endpoints with a list of 'known hacker tool executable names' we can detect this suspicious activity.

Alert Volume

Low (?)

SPL Difficulty

Basic

Stage 3

MITRE ATT&CK Tactics

Discovery | Lateral Movement | Execution

Kill Chain Phases

Exploitation

Data Sources

Endpoint Detection and Response | Windows Security

› Related Splunk Capabilities
› How to Implement
› Known False Positives
› How To Respond
› Show Search
› Help

Outlier(s)

1

Raw Event(s)

11,047

Splunk IT Essentials Learn & Work

Reduce time to identify & resolve infrastructure problems with out-of-the-box searches

Learn how to deploy common IT use cases and apply those use cases in your environment

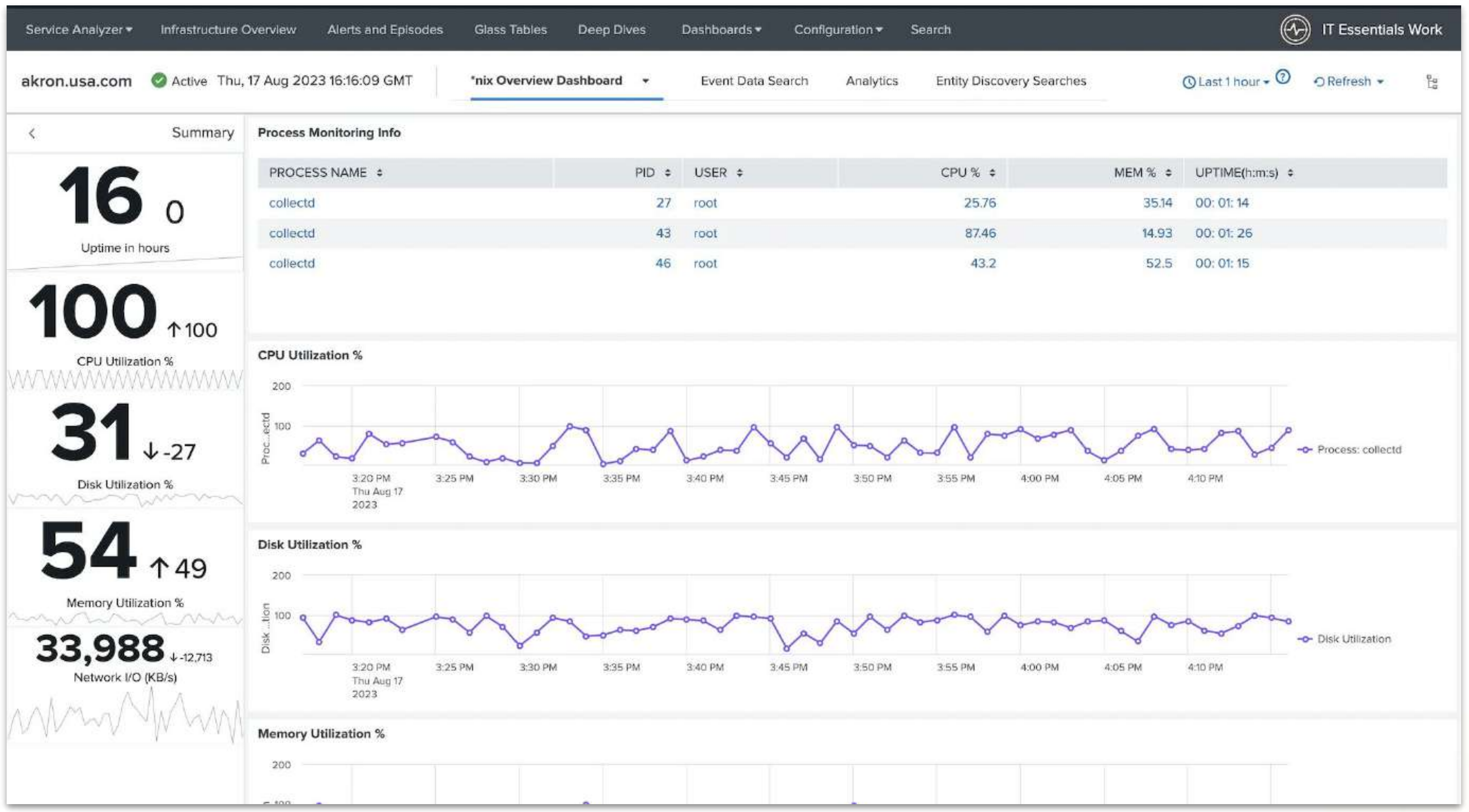
Includes 50+ use cases across:

- Infrastructure Troubleshooting
- Container & Server Monitoring

Outcomes & value:

- Track infrastructure issues in one place
- More efficient IT Operations teams

Get Splunk IT Essentials Learn & Work for free:
https://splunkbase.splunk.com/app/5390/
https://splunkbase.splunk.com/app/5403/



Process Monitoring Info

PROCESS NAME	PID	USER	CPU %	MEM %	UPTIME(h:m:s)
collectd	27	root	25.76	35.14	00: 01: 14
collectd	43	root	87.46	14.93	00: 01: 26
collectd	46	root	43.2	52.5	00: 01: 15

Splunk Infrastructure Monitoring

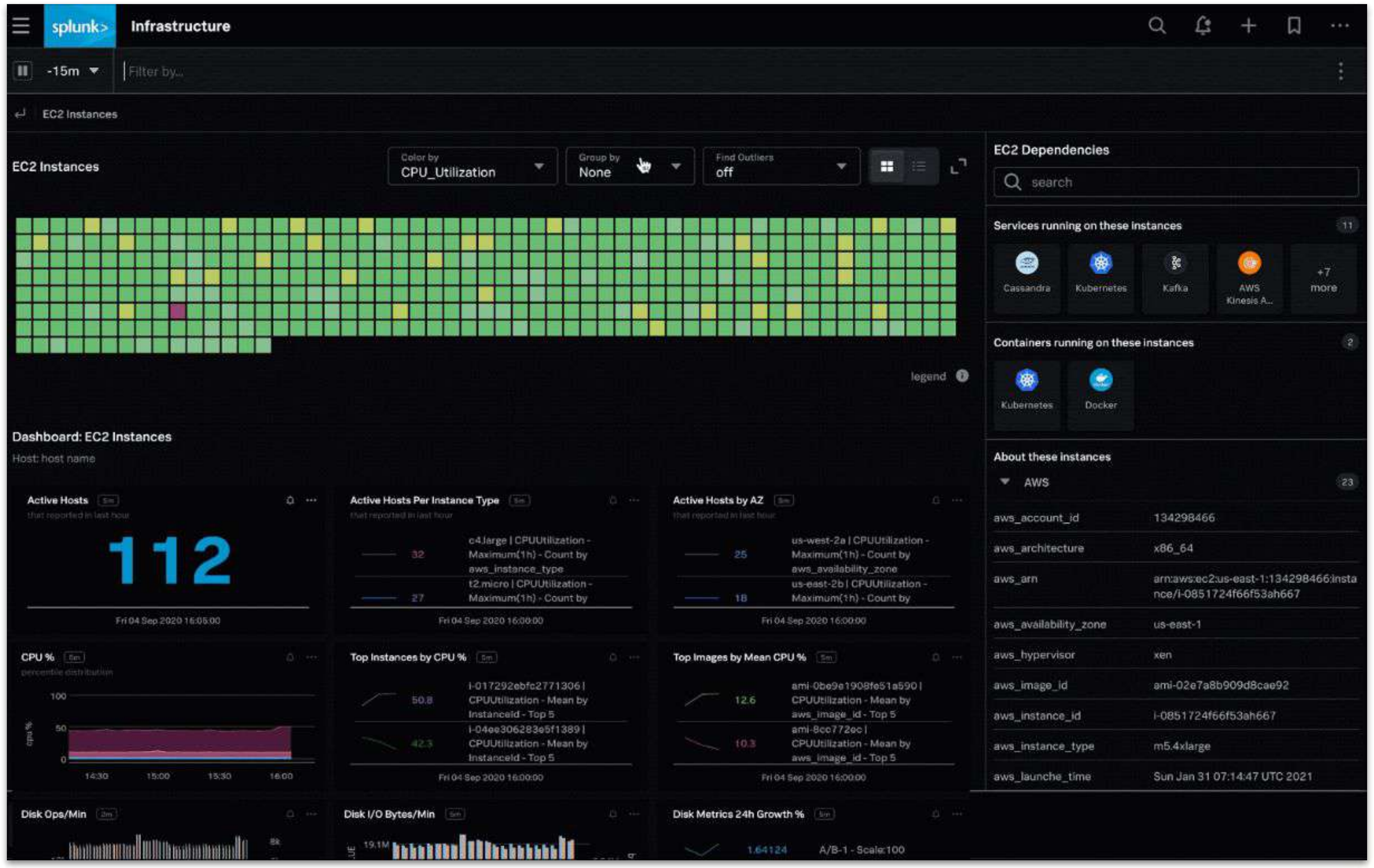
Built on a real-time streaming metrics platform

On-prem to multi cloud monitoring

- Linux, Windows, VMware
- AWS, Google Cloud, Azure
- Kubernetes & Serverless

Automatic service discovery

- 200+ pre-built integrations
- Analytics-driven alerting



Dashboard: EC2 Instances
Host: host name

Active Hosts — 112 — Fri 04 Sep 2020 16:05:00

Active Hosts Per Instance Type
32 c4.large | CPUUtilization - Maximum(1h) - Count by aws_instance_type
27 t2.micro | CPUUtilization - Maximum(1h) - Count by
Fri 04 Sep 2020 16:00:00

Active Hosts by AZ
25 us-west-2a | CPUUtilization - Maximum(1h) - Count by aws_availability_zone
18 us-east-2b | CPUUtilization - Maximum(1h) - Count by
Fri 04 Sep 2020 16:00:00

CPU %

Top Instances by CPU %
50.8 i-017292ebfc2771306 | CPUUtilization - Mean by InstanceId - Top 5
42.3 i-04ee306283e5f1389 | CPUUtilization - Mean by InstanceId - Top 5
Fri 04 Sep 2020 16:00:00

Top Images by Mean CPU %
12.6 ami-0be9e1908fe51a590 | CPUUtilization - Mean by aws_image_id - Top 5
10.3 ami-8cc772ec | CPUUtilization - Mean by aws_image_id - Top 5
Fri 04 Sep 2020 16:00:00

Disk Ops/Min
Disk I/O Bytes/Min
Disk Metrics 24h Growth % — 1.64124 A/B-1 - Scale:100

About these instances
AWS 23

aws_account_id	134298466
aws_architecture	x86_64
aws_arn	arn:aws:ec2:us-east-1:134298466:instance/i-0851724f66f53ah667
aws_availability_zone	us-east-1
aws_hypervisor	xen
aws_image_id	ami-02e7a8b909d8cae92
aws_instance_id	i-0851724f66f53ah667
aws_instance_type	m5.4xlarge
aws_launche_time	Sun Jan 31 07:14:47 UTC 2021

Start monitoring your infrastructure with Splunk IM:
https://www.splunk.com/en_us/download/infrastructure-monitoring.html

Other Splunk Apps on Splunkbase

<u>**Technology Apps**</u>

- Amazon Web Services
- Cisco ASA/IPS/ISE
- DNS Analytics
- Microsoft Exchange
- F5 BIG-IP
- Google Cloud Platform
- Infoblox
- McAfee
- Microsoft Azure
- Microsoft Active Directory
- Microsoft IIS
- Microsoft SQL Server
- Microsoft Windows
- Microsoft Windows DNS
- New Relic
- Unix & Linux
- Oracle
- Palo Alto Networks
- Qualys
- ServiceNow
- Splunk Connect for Kubernetes
- Splunk Connect for Syslog
- VMWare

+ 3000 more!!

Thank you

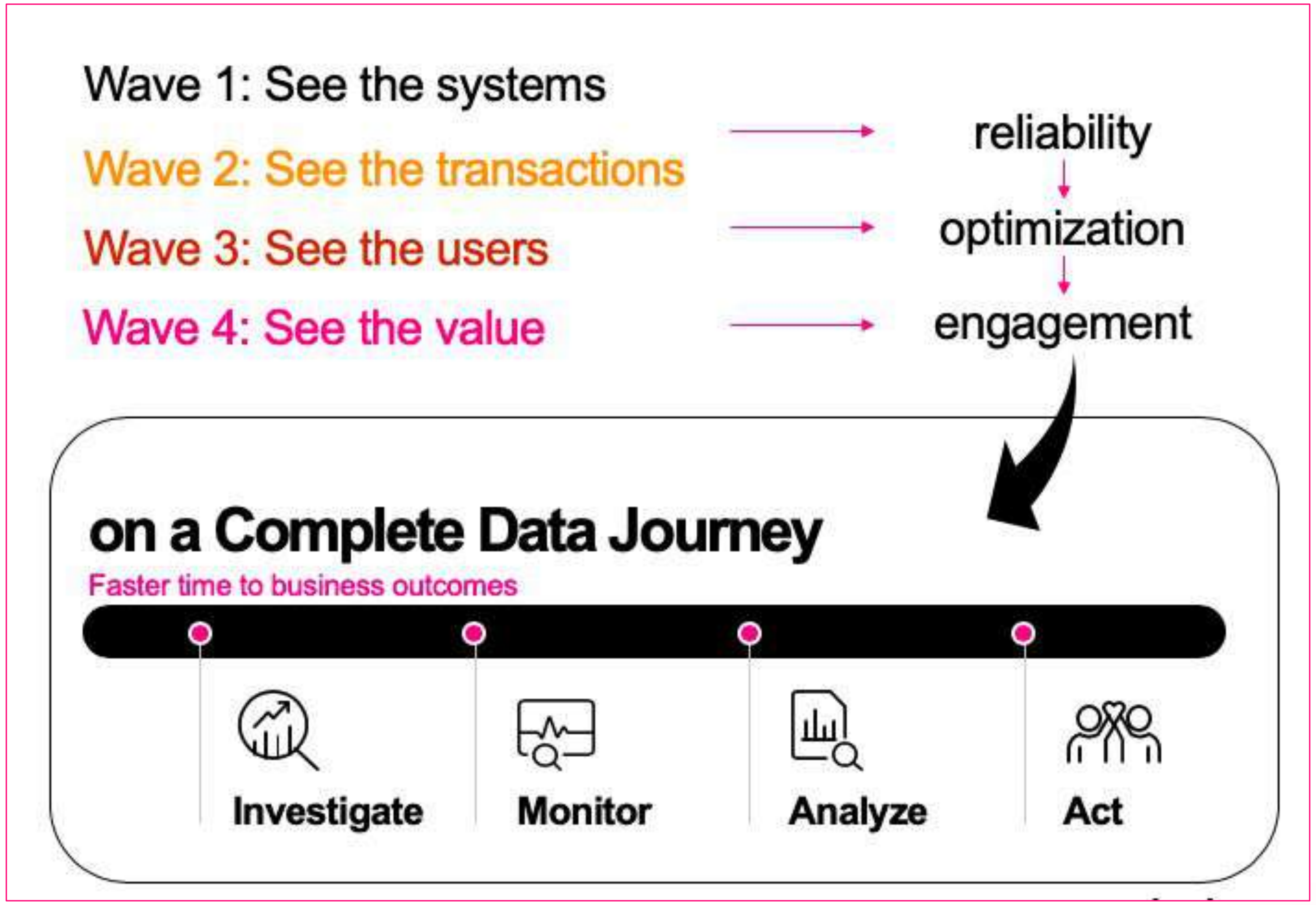
Any questions?

Splunk 201

splunk>

Transactional Analysis

WHY transactional analysis is important

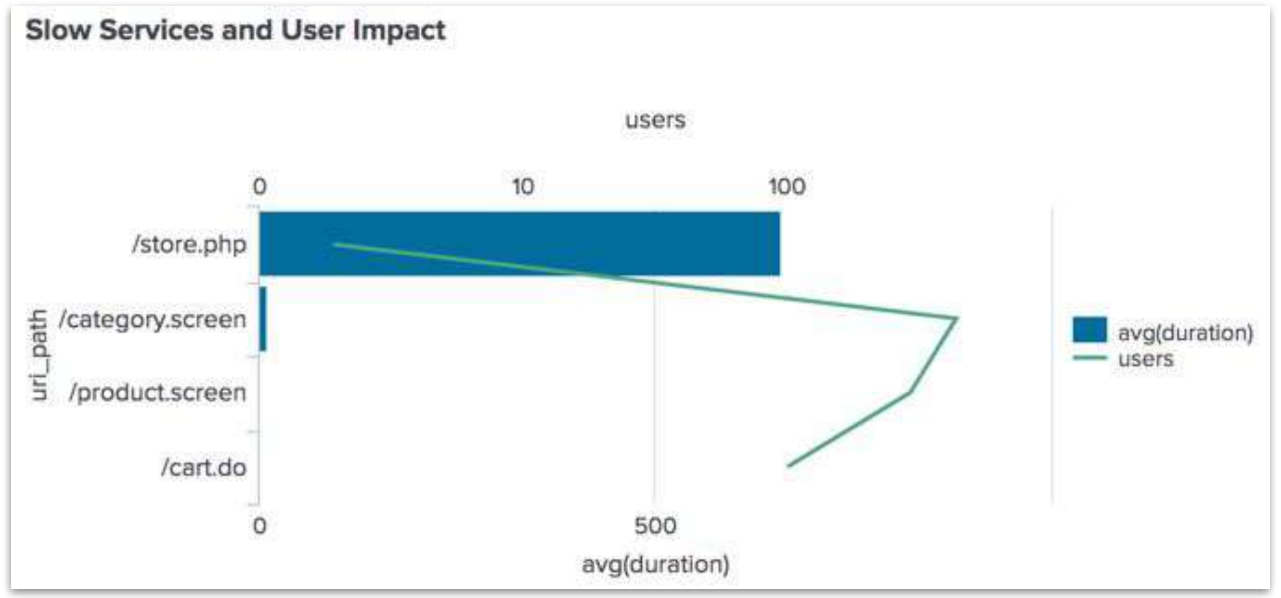


Use cases & value for transactional analysis

Transactional analysis helps you identify higher-level patterns

Use cases:

- Find impacted systems: improve system reliability
- Find impacted transactions: improve % of successful transactions & app performance/reliability
- Find impacted users: improve user engagement and value
- Find impacted process: improve revenue and efficiency



Data and events are the “atoms” of analysis

Use statistical groupings and the transaction command to build system, transactions and user profiles out of granular machine data

Correlate application logs and Application Performance Management (APM) metrics

Build transactions with stats list/values

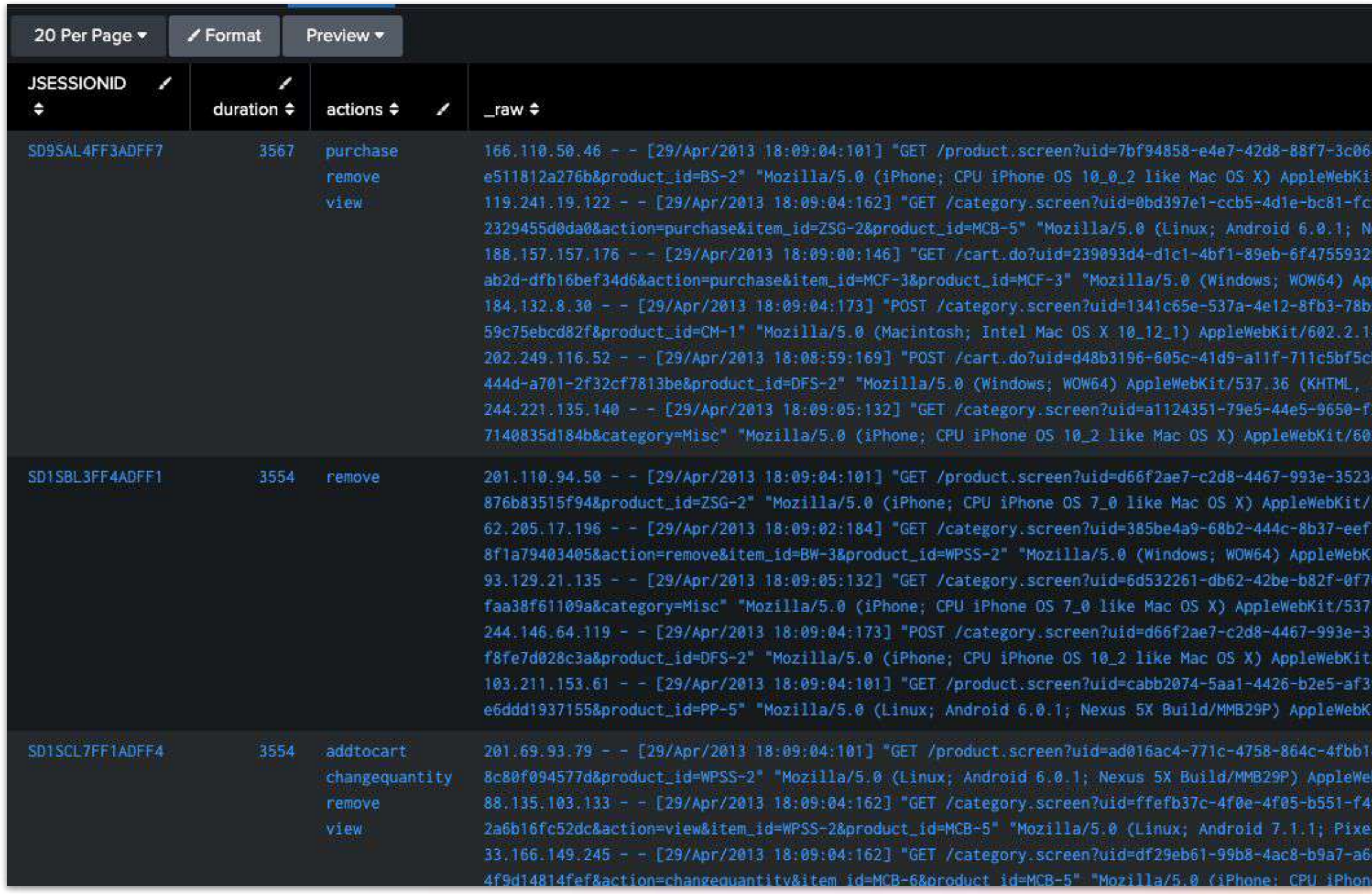
More efficient and optimal with “stats list” or “stats value” commands

Get all details for slowest transactions using stats list or stats values and group by common ID (e.g., username, src, JSESSIONID):

- `range(_time) AS duration` – compute duration of transaction
- `list(X)` – list all values in sequential order
- `values(X)` – list deduplicated values in alphabetical order

Example:

```
sourcetype=access_combined
| fields - address, category_id
| stats range(_time) as duration
  values(action) as actions
  list(_raw) as _raw by JSESSIONID
| sort -duration
| head 10
```


20 Per Page ▾ | Format | Preview ▾

JSESSIONID	duration	actions	_raw
SD9SAL4FF3ADFF7	3567	purchase remove view	166.110.50.46 - - [29/Apr/2013 18:09:04:101] "GET /product.screen?uid=7bf94858-e4e7-42d8-88f7-3c06 e511812a276b&product_id=BS-2" "Mozilla/5.0 (iPhone; CPU iPhone OS 10_0_2 like Mac OS X) AppleWebKi 119.241.19.122 - - [29/Apr/2013 18:09:04:162] "GET /category.screen?uid=0bd397e1-ccb5-4d1e-bc81-fc 2329455d0da0&action=purchase&item_id=ZSG-2&product_id=MCB-5" "Mozilla/5.0 (Linux; Android 6.0.1; N 188.157.157.176 - - [29/Apr/2013 18:09:00:146] "GET /cart.do?uid=239093d4-d1c1-4bf1-89eb-6f4755932 ab2d-dfb16bef34d6&action=purchase&item_id=MCF-3&product_id=MCF-3" "Mozilla/5.0 (Windows; WOW64) Ap 184.132.8.30 - - [29/Apr/2013 18:09:04:173] "POST /category.screen?uid=1341c65e-537a-4e12-8fb3-78b 59c75ebcd82f&product_id=CM-1" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_12_1) AppleWebKit/602.2.1 202.249.116.52 - - [29/Apr/2013 18:08:59:169] "POST /cart.do?uid=d48b3196-605c-41d9-a11f-711c5bf5c 444d-a701-2f32cf7813be&product_id=DFS-2" "Mozilla/5.0 (Windows; WOW64) AppleWebKit/537.36 (KHTML, 244.221.135.140 - - [29/Apr/2013 18:09:05:132] "GET /category.screen?uid=a1124351-79e5-44e5-9650-f 7140835d184b&category=Misc" "Mozilla/5.0 (iPhone; CPU iPhone OS 10_2 like Mac OS X) AppleWebKit/60
SD1SBL3FF4ADFF1	3554	remove	201.110.94.50 - - [29/Apr/2013 18:09:04:101] "GET /product.screen?uid=d66f2ae7-c2d8-4467-993e-3523 876b83515f94&product_id=ZSG-2" "Mozilla/5.0 (iPhone; CPU iPhone OS 7_0 like Mac OS X) AppleWebKit/ 62.205.17.196 - - [29/Apr/2013 18:09:02:184] "GET /category.screen?uid=385be4a9-68b2-444c-8b37-eef 8f1a79403405&action=remove&item_id=BW-3&product_id=WPSS-2" "Mozilla/5.0 (Windows; WOW64) AppleWebK 93.129.21.135 - - [29/Apr/2013 18:09:05:132] "GET /category.screen?uid=6d532261-db62-42be-b82f-0f7 faa38f61109a&category=Misc" "Mozilla/5.0 (iPhone; CPU iPhone OS 7_0 like Mac OS X) AppleWebKit/537 244.146.64.119 - - [29/Apr/2013 18:09:04:173] "POST /category.screen?uid=d66f2ae7-c2d8-4467-993e-3 f8fe7d028c3a&product_id=DFS-2" "Mozilla/5.0 (iPhone; CPU iPhone OS 10_2 like Mac OS X) AppleWebKit 103.211.153.61 - - [29/Apr/2013 18:09:04:101] "GET /product.screen?uid=cabb2074-5aa1-4426-b2e5-af3 e6ddd1937155&product_id=PP-5" "Mozilla/5.0 (Linux; Android 6.0.1; Nexus 5X Build/MMB29P) AppleWebK
SD1SCL7FF1ADFF4	3554	addtocart changequantity remove view	201.69.93.79 - - [29/Apr/2013 18:09:04:101] "GET /product.screen?uid=ad016ac4-771c-4758-864c-4fbb1 8c80f094577d&product_id=WPSS-2" "Mozilla/5.0 (Linux; Android 6.0.1; Nexus 5X Build/MMB29P) AppleWe 88.135.103.133 - - [29/Apr/2013 18:09:04:162] "GET /category.screen?uid=ffefb37c-4f0e-4f05-b551-f4 2a6b16fc52dc&action=view&item_id=WPSS-2&product_id=MCB-5" "Mozilla/5.0 (Linux; Android 7.1.1; Pixe 33.166.149.245 - - [29/Apr/2013 18:09:04:162] "GET /category.screen?uid=df29eb61-99b8-4ac8-b9a7-a6 4f9d14814fef&action=changequantity&item_id=MCB-6&product_id=MCB-5" "Mozilla/5.0 (iPhone; CPU iPhon

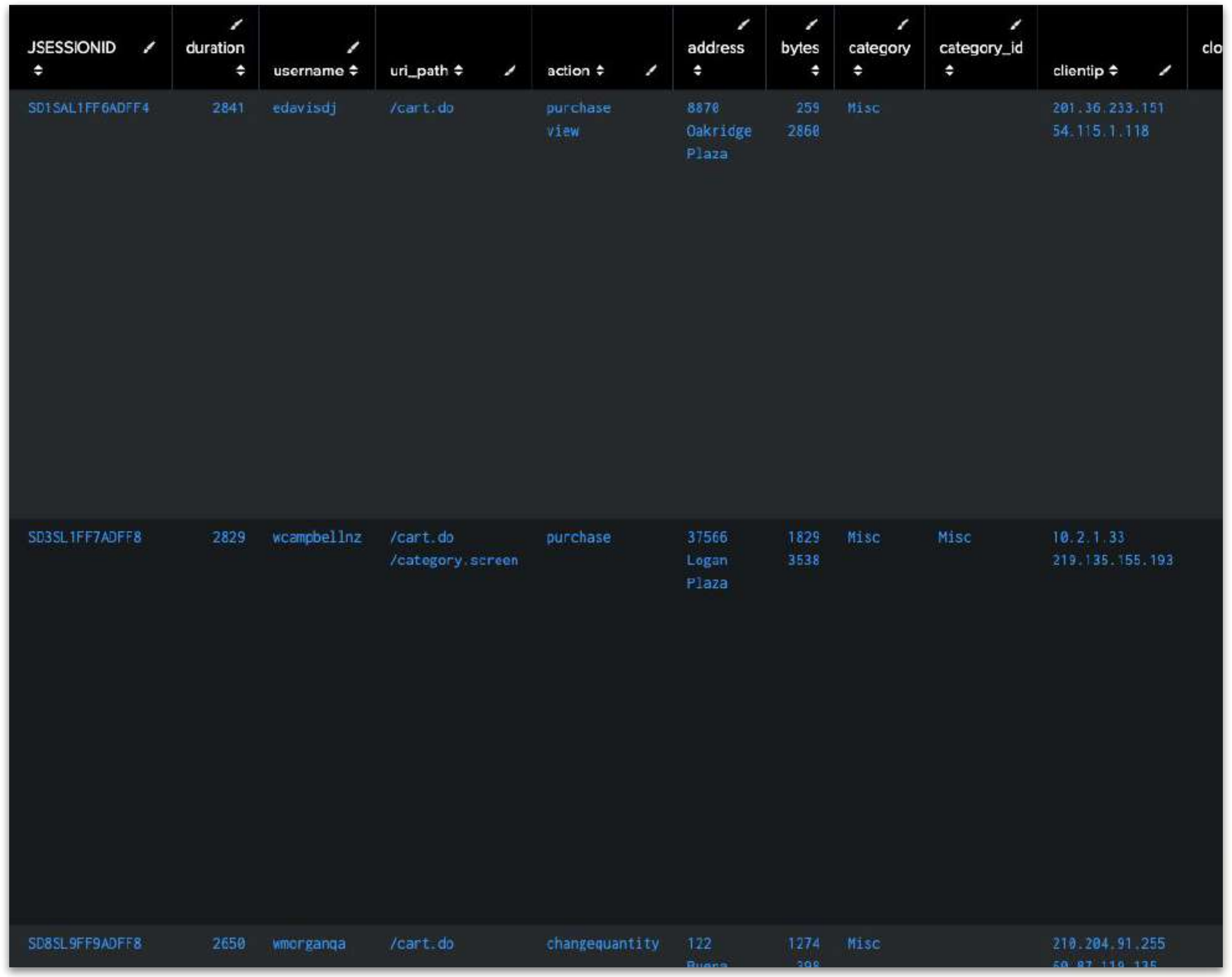
Build transactions with transaction command

Less efficient but easy with “transaction” command
Get all details for slowest transactions using transaction command

```
sourcetype=access_combined
| transaction JSESSIONID username
| sort -duration
| head 10
| table JSESSIONID duration username uri_path *
```

Warning: the “transaction” command is not efficient!

- Use the “stats list” or “stats value” commands if you can
- Talk to your Splunk admin team
- Read more: https://docs.splunk.com/Documentation/Splunk/latest/SearchReference/Transaction



JSESSIONID	duration	username	uri_path	action	address	bytes	category	category_id	clientip	clo
SD1SAL1FF6ADFF4	2841	edavisdj	/cart.do	purchase view	8870 Oakridge Plaza	259 2860	Misc		201.36.233.151 54.115.1.118	
SD3SL1FF7ADFF8	2829	wcampbellnz	/cart.do /category.screen	purchase	37566 Logan Plaza	1829 3538	Misc	Misc	10.2.1.33 219.135.155.193	
SD8SL9FF9ADFF8	2650	wmorganqa	/cart.do	changequantity	122	1274	Misc		210.204.91.255	

`stats list/values` vs `transaction` command

`stats list` and `stats values` commands are good in most circumstances:

- Efficient transactions
- Transactions with common correlation fields or transaction ID
- `stats list` – all values, sequential
- `stats values` – deduplicated values, alphabetical

transaction command can be used in a pinch:

- Quick prototypes
- Complex transactions (startswith, endswith, maxspan)

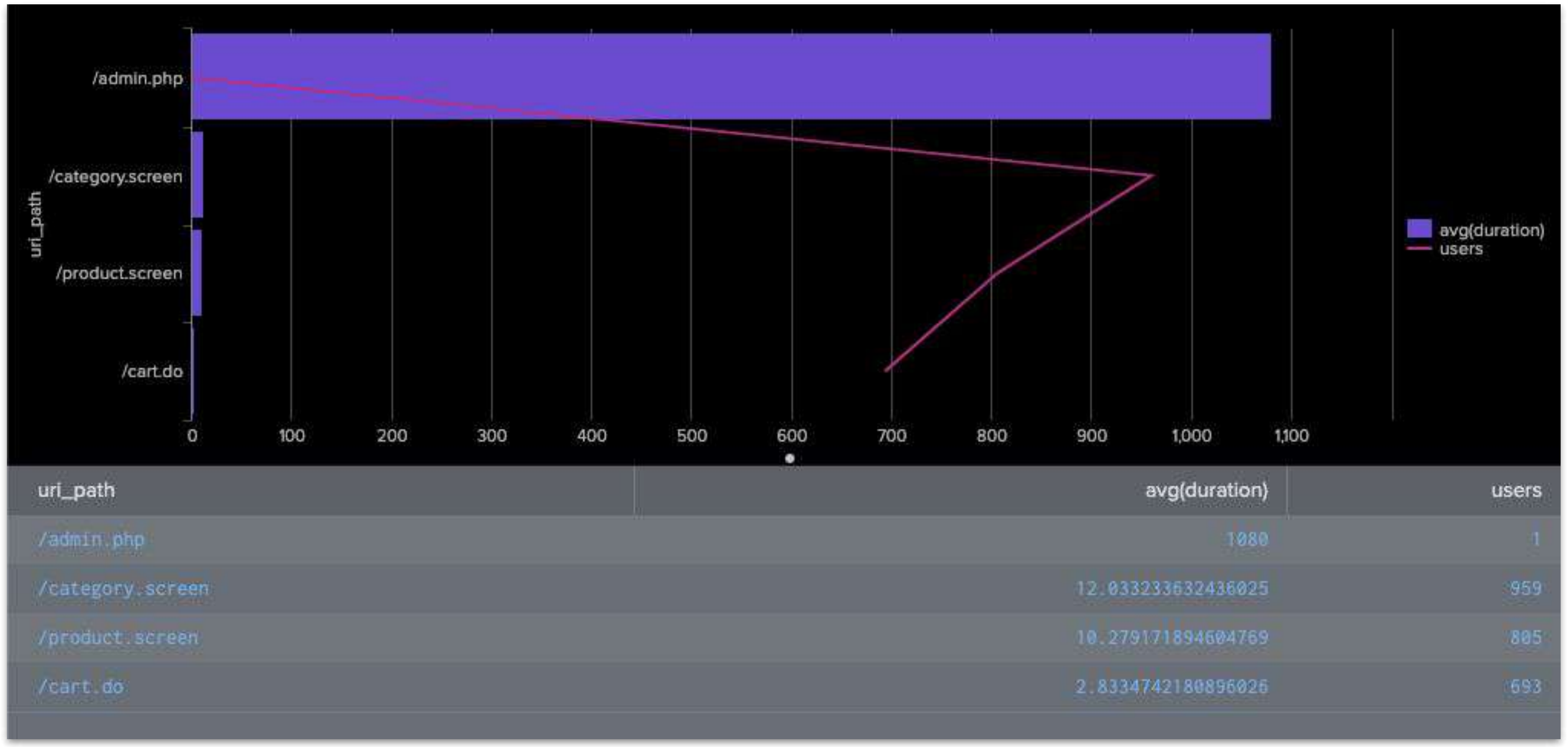
Warning: the “transaction” command is not efficient!

- Use the “stats list” or “stats value” commands if you can
- Talk to your Splunk admin team

Analyze transactions with additional stats commands

Analyze transactions with additional aggregations:

- count
- sum
- avg / stdev
- median / perc
- dc / estdc



uri_path	avg(duration)	users
/admin.php	1080	1
/category.screen	12.033233632436025	959
/product.screen	10.279171894604769	805
/cart.do	2.8334742180896026	693

Example: Find slow services and user impact by aggregating transactions:

```
sourcetype=access_combined
| stats values(action) as actions range(_time) as duration
  by JSESSIONID username src uri_path
| stats avg(duration) dc(username) as users by uri_path
| sort - "avg(duration)"
```

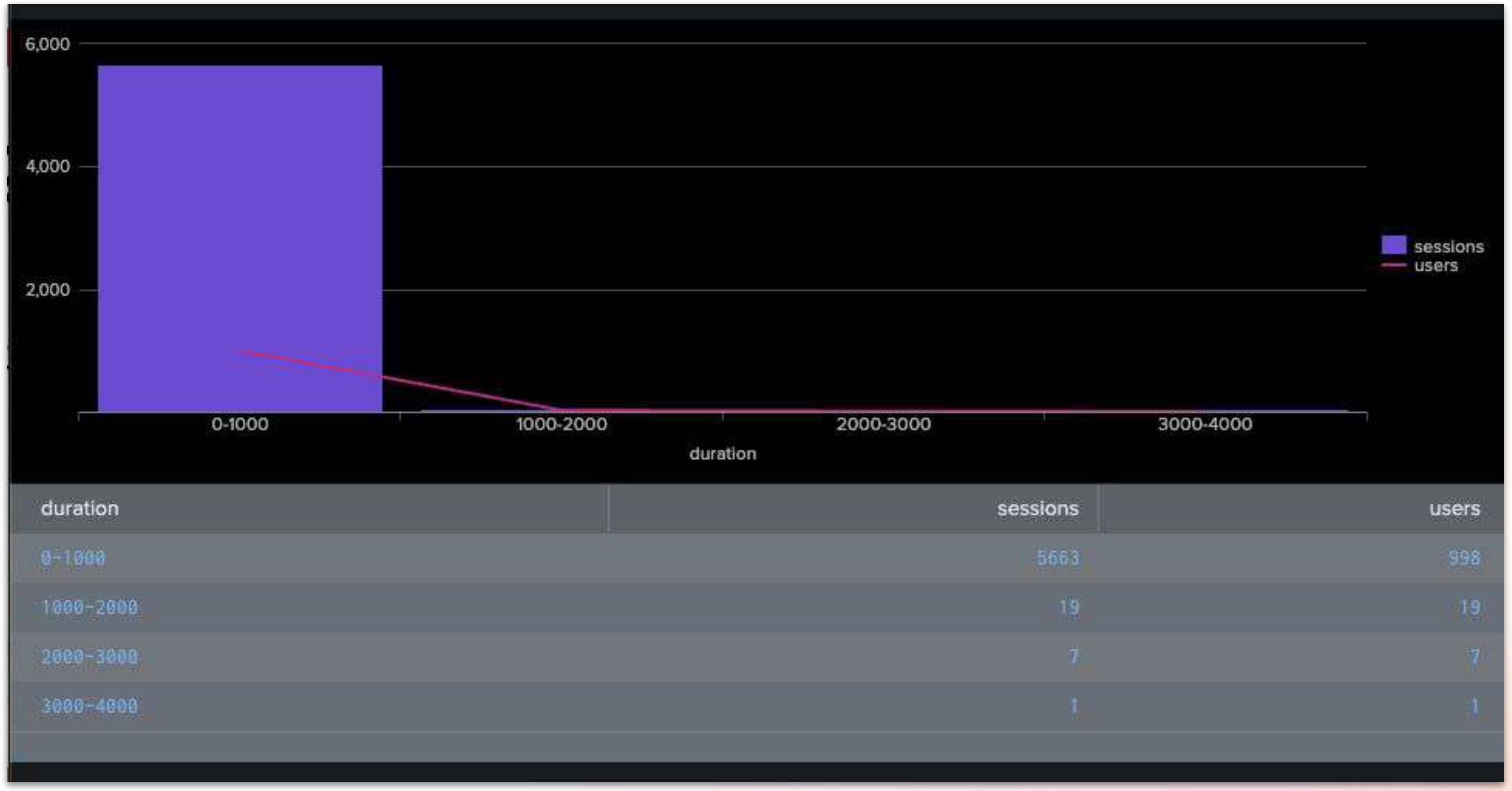
Monitor systems with histograms & chart command

Compute histograms for monitoring real-time transactions

Use chart command with bins parameter, or standalone bin command

Monitor transaction speeds and user impact:

```
sourcetype=access_combined
| stats range(_time) as duration
    by JSESSIONID username
| chart count as sessions
    dc(username) as users
    by duration bins=10
```



duration	sessions	users
0-1000	5663	998
1000-2000	19	19
2000-3000	7	7
3000-4000	1	1

Advanced Statistics

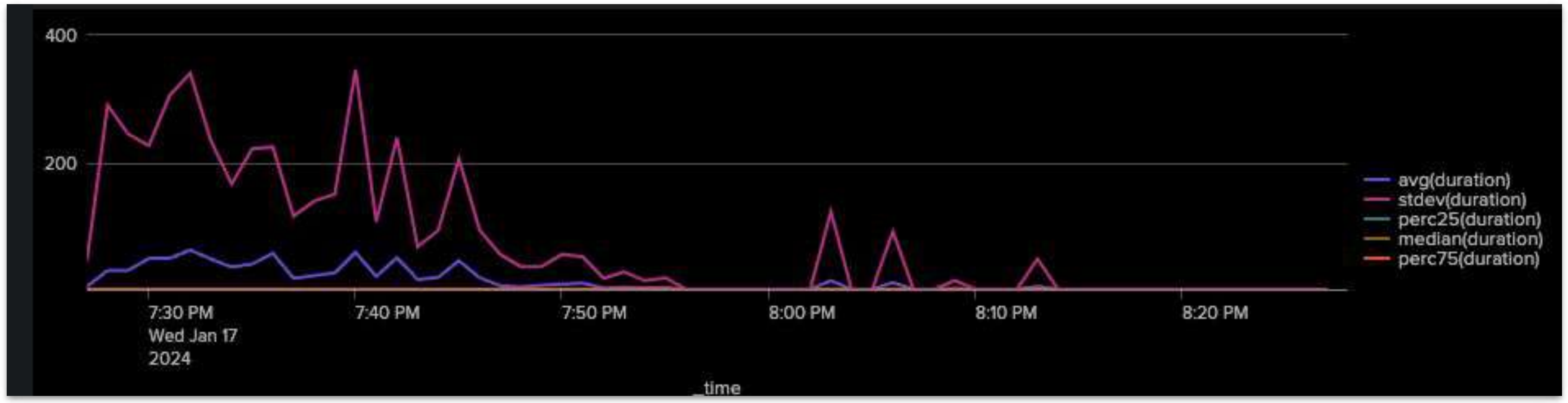
splunk>

Comparing statistics over time

Track the "typical" value using averages and medians. Track the fluctuations using stdev and percentiles. If there are problems, likely to modify either typical values or fluctuations.

Compare different transaction speeds over time:

```
sourcetype=access_combined
| stats earliest(_time) as _time range(_time) as duration by JSESSIONID username
| timechart avg(duration) stdev(duration) perc25(duration) median(duration) perc75(duration)
```



Advanced Statistics vs Machine Learning

Splunk makes it easy to do multiple types of statistics:

- **(101) Statistics:** aggregate and analyze numerical data split by groups (counts, sums, avgs & medians)
- **(201) Transactional Statistics:** analyze higher-level entities and transactions from raw data
- **(201) Advanced Statistics:** incorporate variability (stdev/percentiles), apply eventstats & streamstats
- **(301) Machine Learning:** use statistical models to detect anomalies and sudden changes, adjust thresholds dynamically, and predict service degradation

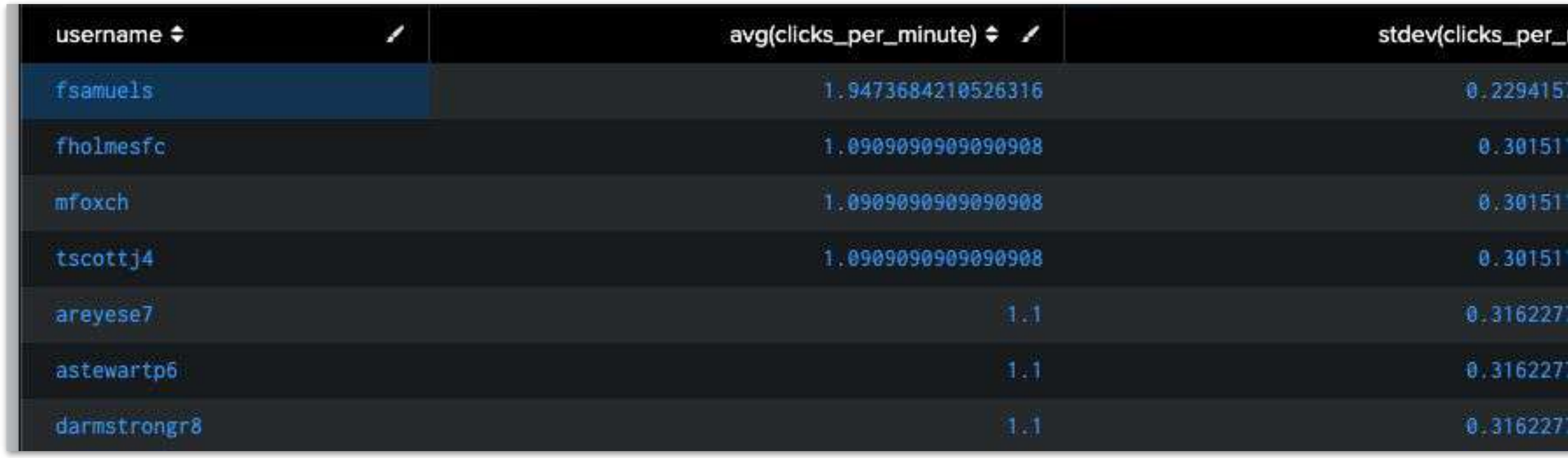
References:

- List of stats functions: https://docs.splunk.com/Documentation/Splunk/latest/SearchReference/CommonStatsFunctions
- Download the Splunk Machine Learning Toolkit (MLTK): https://splunkbase.splunk.com/app/2890/
- Download Python for Scientific Computing add-on for your OS: https://splunkbase.splunk.com/app/2882

Use standard deviation to find anomalies

Apply standard deviation and variance to transactions like sessions and users.

This is very useful when looking for bots or looking at trends across segments of users



username	avg(clicks_per_minute)	stdev(clicks_per_
fsamuels	1.9473684210526316	0.229415
fholmesfc	1.0909090909090908	0.30151
mfoxch	1.0909090909090908	0.30151
tscottj4	1.0909090909090908	0.30151
areyese7	1.1	0.316227
astewartp6	1.1	0.316227
darmstrongr8	1.1	0.316227

Example: Detect bots in web logs (very low standard deviation, constant clicks/minute)

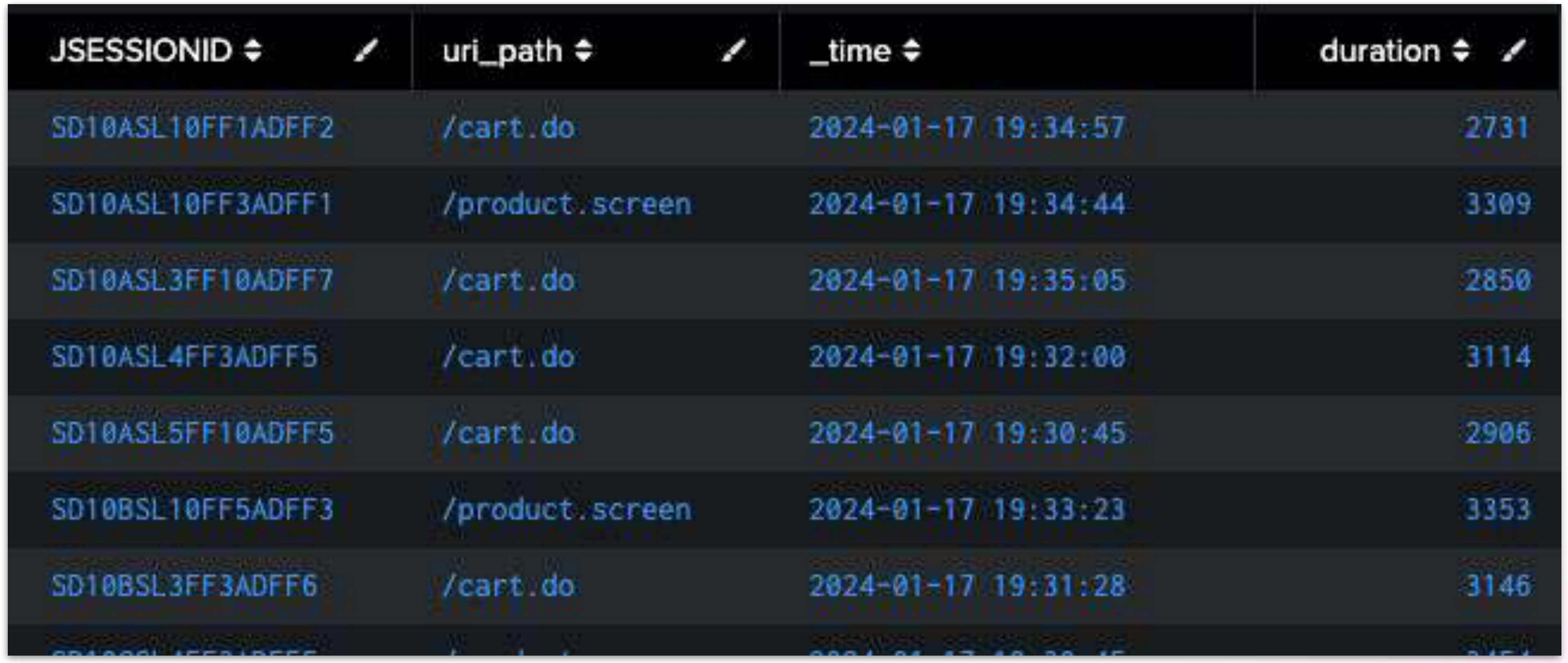
```
sourcetype=access_combined
| bin _time span=1m
| stats count as clicks_per_minute by _time username
| stats avg(clicks_per_minute) stdev(clicks_per_minute) by username
| where 'avg(clicks_per_minute)' > 1
| sort + "stdev(clicks_per_minute)"
```

Peer-based adaptive thresholds with eventstats

Use eventstats to build adaptive thresholds based on peer groups (services, application tags, business units), and and track service health over time.

Example: Find all transactions with very slow durations, with variable speed impact by service

```
sourcetype=access_combined
| stats earliest(_time) as _time range(_time) as duration by JSESSIONID uri_path
| eventstats avg(duration) stdev(duration) by uri_path
| where duration > 'avg(duration)' + 2*'stdev(duration)'
```



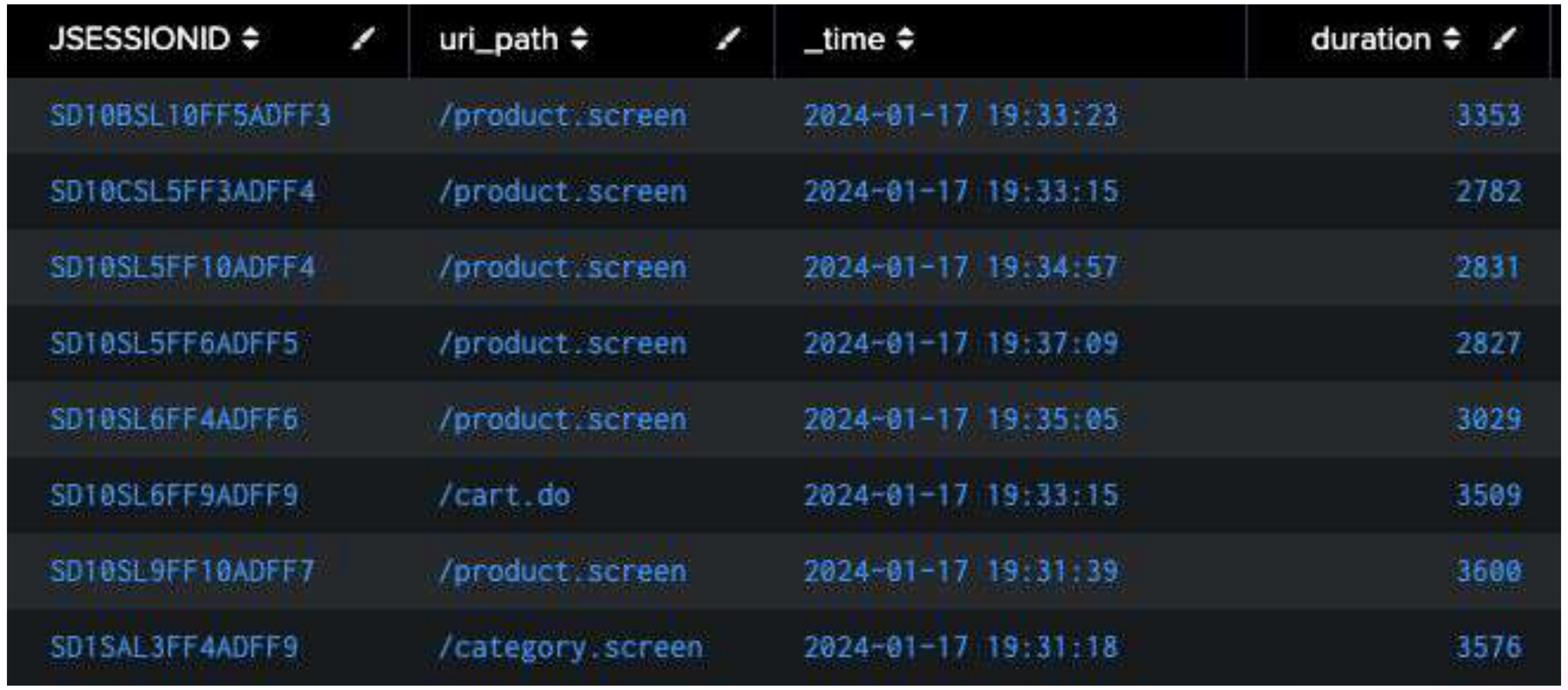
JSESSIONID	uri_path	_time	duration
SD10ASL10FF1ADFF2	/cart.do	2024-01-17 19:34:57	2731
SD10ASL10FF3ADFF1	/product.screen	2024-01-17 19:34:44	3309
SD10ASL3FF10ADFF7	/cart.do	2024-01-17 19:35:05	2850
SD10ASL4FF3ADFF5	/cart.do	2024-01-17 19:32:00	3114
SD10ASL5FF10ADFF5	/cart.do	2024-01-17 19:30:45	2906
SD10BSL10FF5ADFF3	/product.screen	2024-01-17 19:33:23	3353
SD10BSL3FF3ADFF6	/cart.do	2024-01-17 19:31:28	3146

Historical adaptive thresholds with streamstats

Use streamstats to build historical adaptive thresholds and track service health over time.

Find anomalous transactions which are impacting service health (compute rolling avg and stdev of duration by service, find durations which are 2 stdevs slower than historical avg)

```
sourcetype=access_combined
| stats earliest(_time) as _time range(_time) as duration by JSESSIONID uri_path
| streamstats avg(duration) stdev(duration) by uri_path
| where duration > 'avg(duration)' + 2*'stdev(duration)'
```

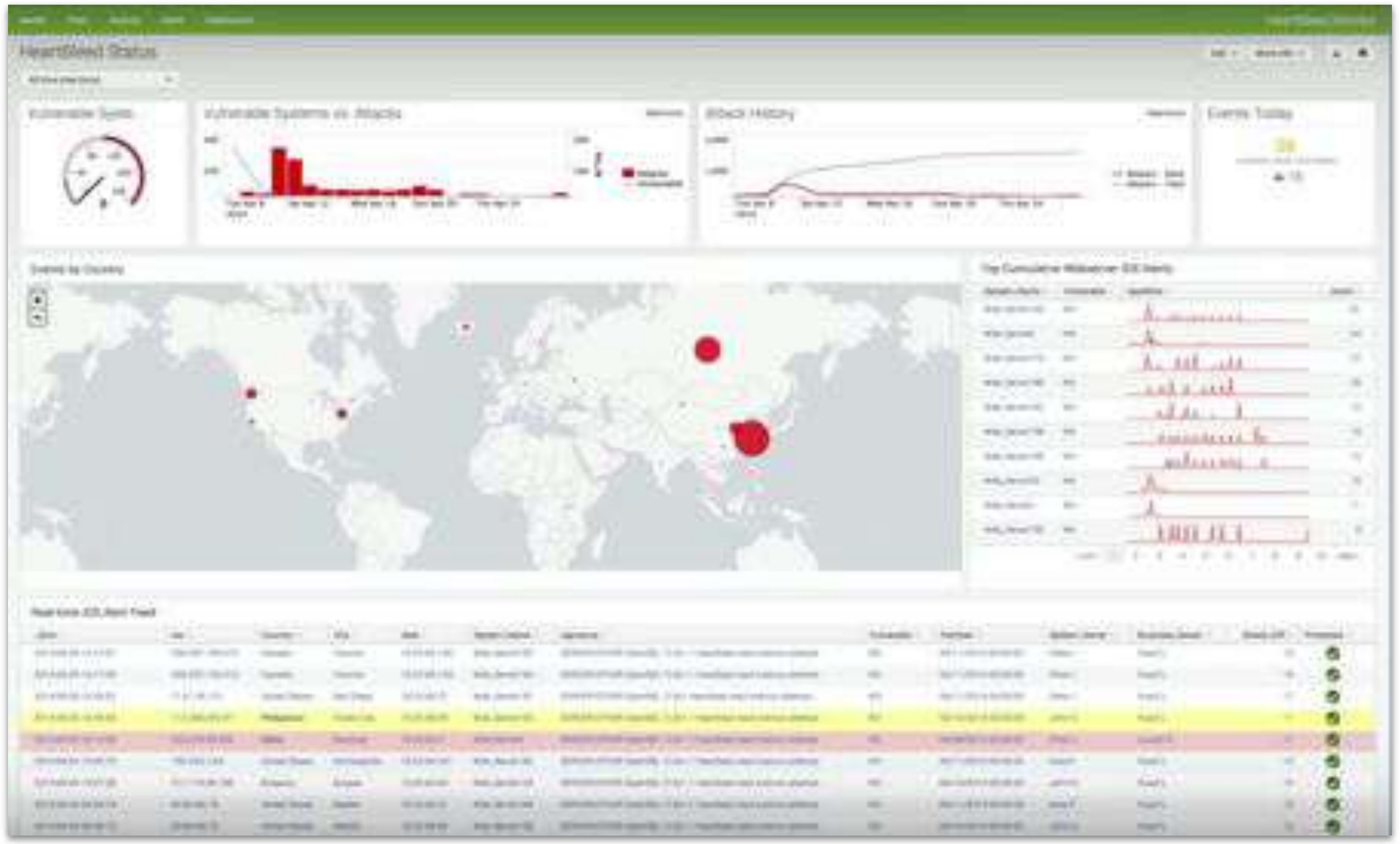


JSESSIONID	uri_path	_time	duration
SD10BSL10FF5ADFF3	/product.screen	2024-01-17 19:33:23	3353
SD10CSL5FF3ADFF4	/product.screen	2024-01-17 19:33:15	2782
SD10SL5FF10ADFF4	/product.screen	2024-01-17 19:34:57	2831
SD10SL5FF6ADFF5	/product.screen	2024-01-17 19:37:09	2827
SD10SL6FF4ADFF6	/product.screen	2024-01-17 19:35:05	3029
SD10SL6FF9ADFF9	/cart.do	2024-01-17 19:33:15	3509
SD10SL9FF10ADFF7	/product.screen	2024-01-17 19:31:39	3600
SD1SAL3FF4ADFF9	/category.screen	2024-01-17 19:31:18	3576

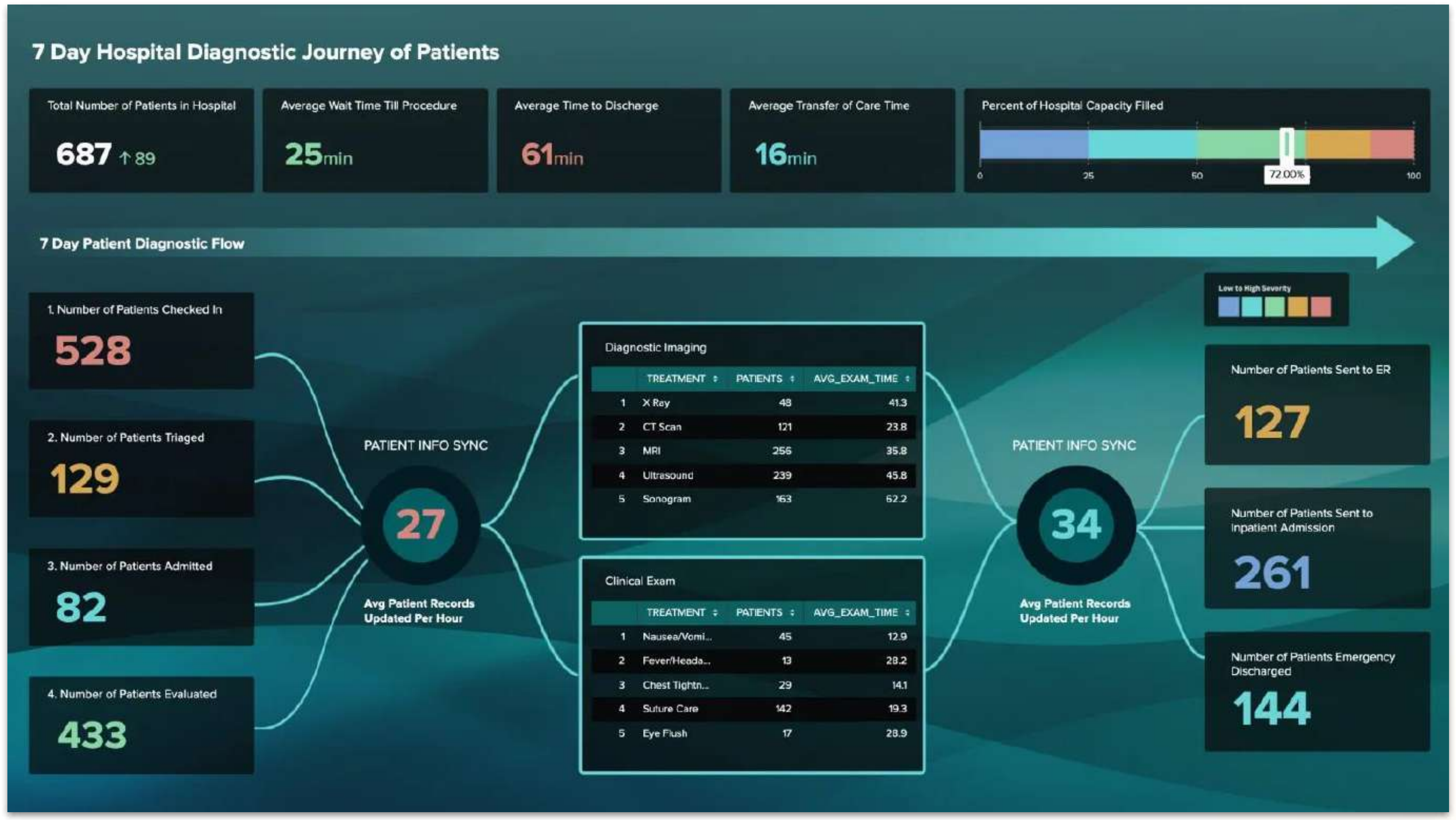
Advanced Dashboarding

Examples of Good Dashboards

NASDAQ's Heartbleed Status dashboard for CISO/CIO:



Dashboard showing patient diagnostics:



7 Day Hospital Diagnostic Journey of Patients

Total Number of Patients in Hospital	Average Wait Time Till Procedure	Average Time to Discharge	Average Transfer of Care Time	Percent of Hospital Capacity Filled
687 ↑ 89	25min	61min	16min	72.00%

7 Day Patient Diagnostic Flow

1. Number of Patients Checked In 528

2. Number of Patients Triaged 129

3. Number of Patients Admitted 82

4. Number of Patients Evaluated 433

PATIENT INFO SYNC 27 Avg Patient Records Updated Per Hour

Diagnostic Imaging

	TREATMENT	PATIENTS	AVG_EXAM_TIME
1	X Ray	48	41.3
2	CT Scan	121	23.8
3	MRI	256	35.8
4	Ultrasound	239	45.8
5	Sonogram	163	62.2

Clinical Exam

	TREATMENT	PATIENTS	AVG_EXAM_TIME
1	Nausea/Vomi...	45	12.9
2	Fever/Heada...	13	28.2
3	Chest Tightn...	29	14.1
4	Suture Care	142	19.3
5	Eye Flush	17	28.9

PATIENT INFO SYNC 34 Avg Patient Records Updated Per Hour

Low to High Severity

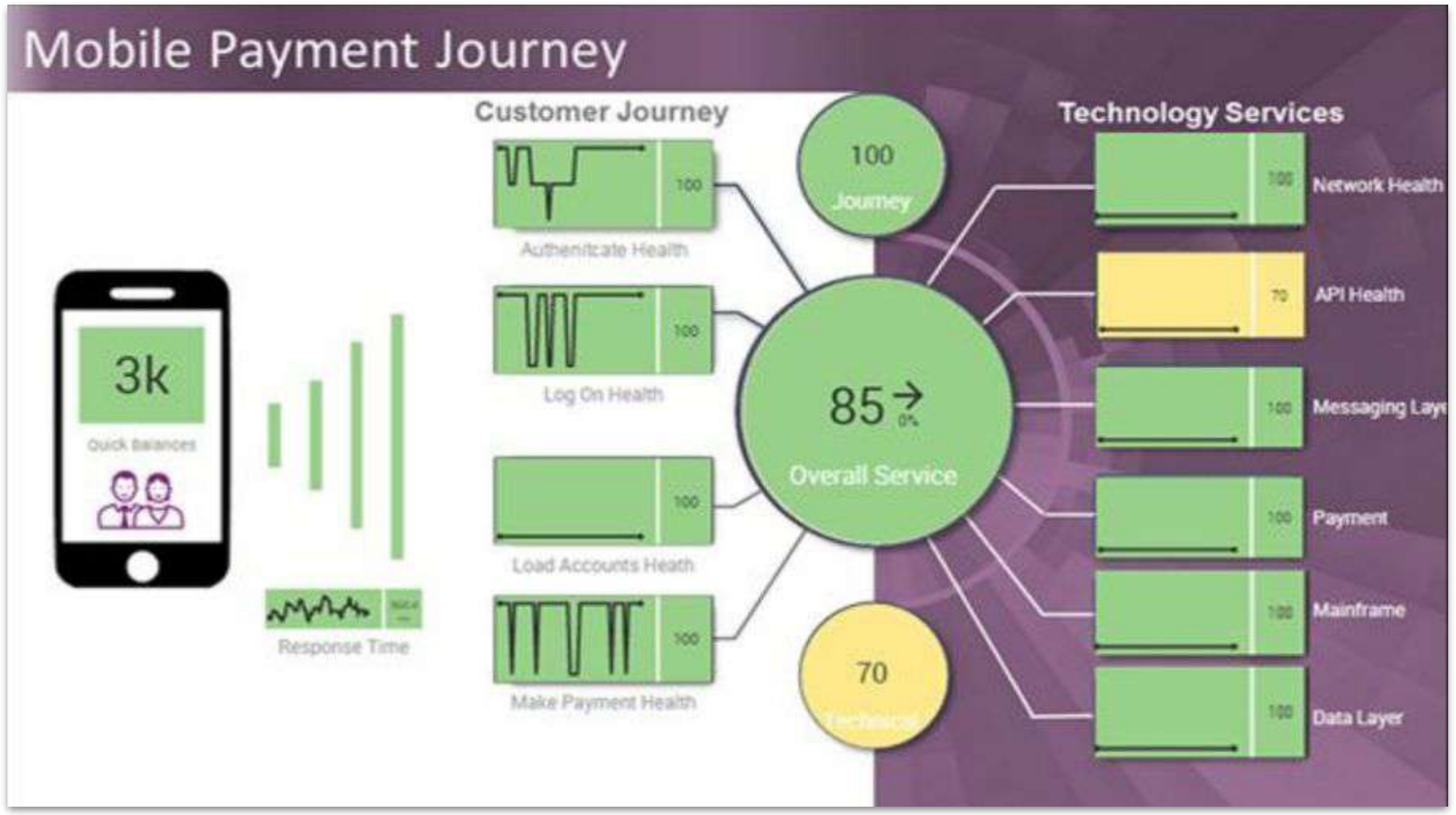
Number of Patients Sent to ER 127

Number of Patients Sent to Inpatient Admission 261

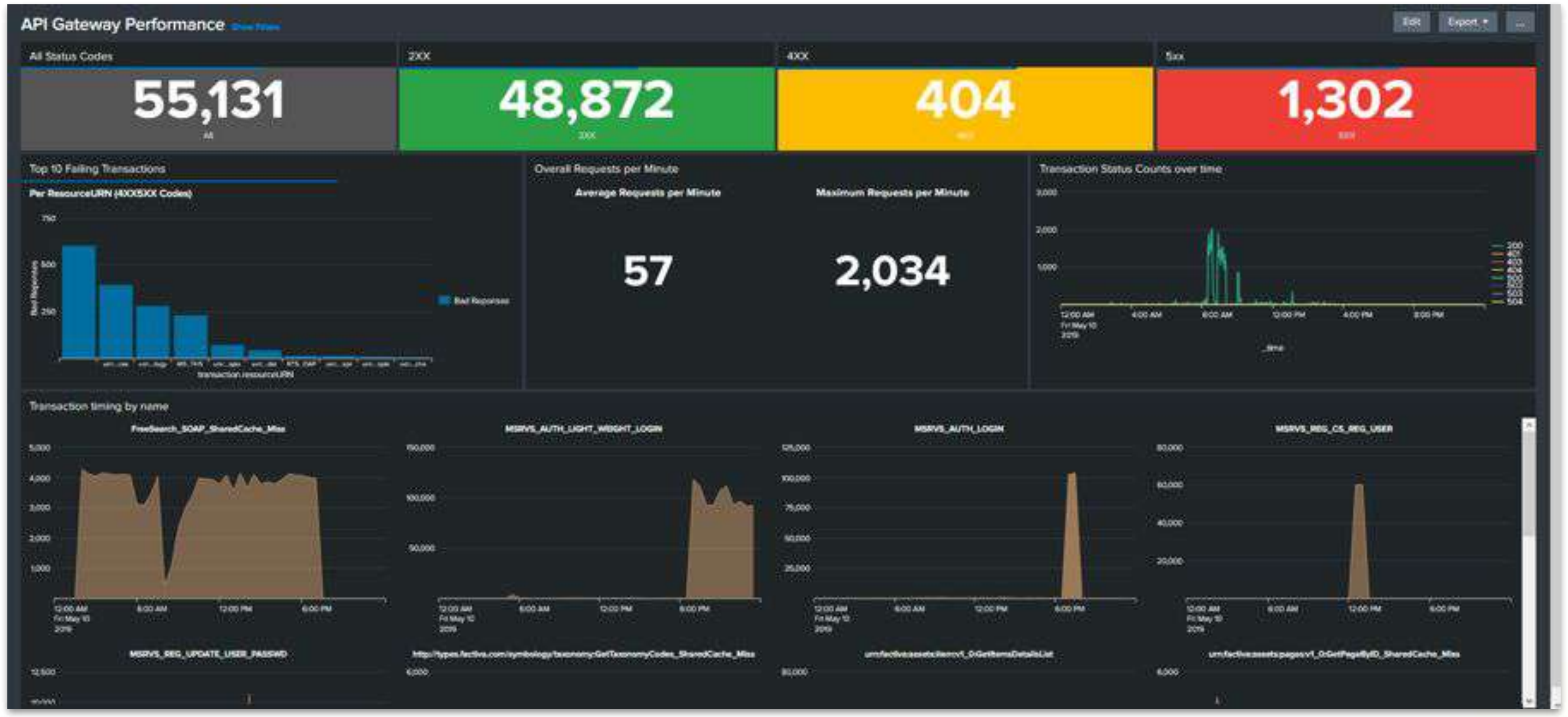
Number of Patients Emergency Discharged 144

Examples of Good Dashboards (cont.)

Allied Irish Bank's Mobile Payment Journey:



API Gateway Performance and Health:


API Gateway Performance
All Status Codes
55,131
2XX
48,872
4XX
404
5xx
1,302
Top 10 Failing Transactions
Overall Requests per Minute
Average Requests per Minute
57
Maximum Requests per Minute
2,034
Transaction Status Counts over time
Transaction timing by name

Dashboarding Best Practices

Understand Requirements

- **Document** stakeholders' needs and context
- **Align** your strategy & metrics to the Business Goals
- **Be prepared** with the key projects, the most vital ones needing approval, and strategy in simple numbers. E.g., "if we go SaaS, reduce costs by $5 million"
- **Don't** build your dashboard in a vacuum

Design Dashboard

- **Define** required charts and metrics
- **Show** Single Values 💯 and Trending Charts 📈 before showing Raw Data 📋
- **Build** a simple prototype/minimum viable product, then iterate with stakeholder.
- **Create** a dashboard in the Splunk app for the population that will view it (e.g., Trading dashboard in Trading team's app workspace)

Tell a Story

- **Focus** on what your audience NEEDS to see, not what's easy to calculate
- **Use** Real Life examples and stories that are contextually relevant to the business
- **Provide** to stats and competing organizations is helpful.

Dashboarding Discussion & Homework

- What is the best dashboard you’ve seen?
- Why was it so good?
- Who was the stakeholder?
- What value did it give them?

Form Inputs and Tokens

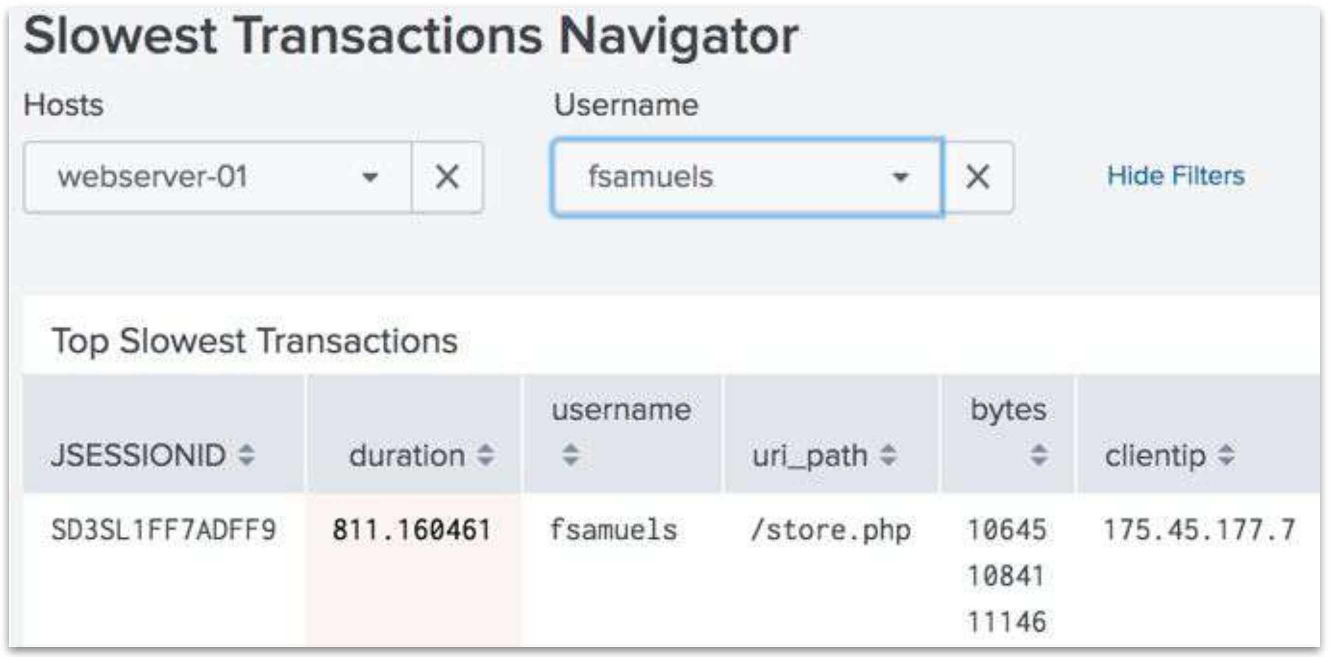
Make dashboards easy to use with form inputs

E.g., split by time, service/application, user, host

Add a form input

- For time input: adjust the time picker on each panel
- For other inputs: use token syntax in the panels:

```
search host=$host$ username=$username$
```



Slowest Transactions Navigator

Hosts: webserver-01

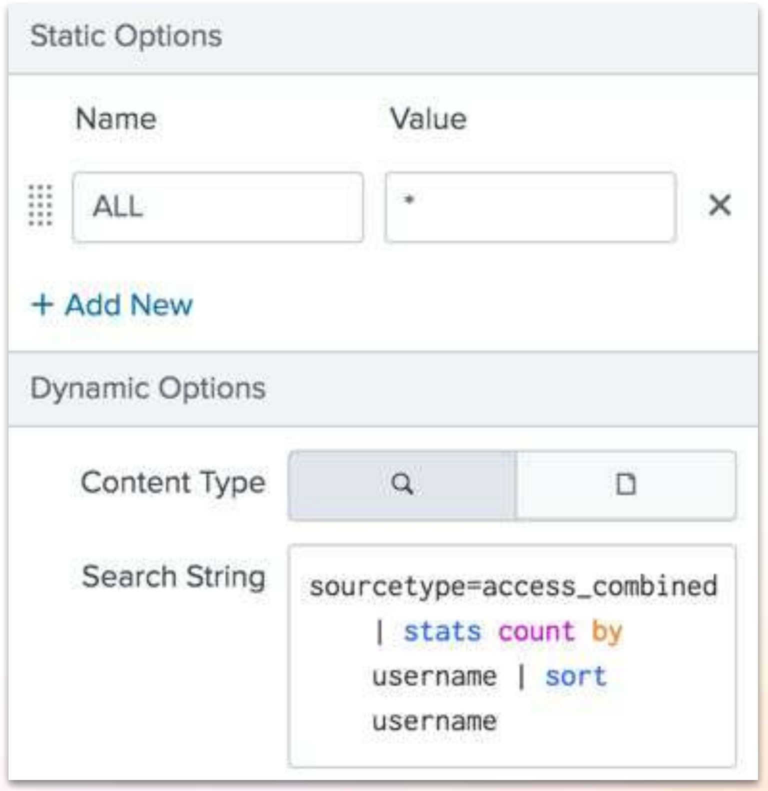
Username: fsamuels

Top Slowest Transactions

JSESSIONID	duration	username	uri_path	bytes	clientip
SD3SL1FF7ADFF9	811.160461	fsamuels	/store.php	10645 10841 11146	175.45.177.7

Best practice:

- Use Static Options for small number of choices, and ALL:
  - Name: ALL
  - Value: *
- Use Dynamic Search to populate data-driven set of choices:
  - `| tstats count where sourcetype=access_combined by host | sort host`
  - `sourcetype=access_combined | stats count by username | sort username`



Static Options

Name	Value
ALL	*

Dynamic Options

Search String:

```
sourcetype=access_combined
| stats count by
username | sort
username
```

Post Process Search

Build “base searches” (with id tag) on your dashboard which pull the data, then “post-process searches” (with base tag) which process from same base dataset.

Best practice: Place values and charts above data and use shared time picker

Step 1) start with a base search generating transaction:

```
<search id="all_web_sessions">
   <query>sourcetype=access*  | stats earliest(_time) as _time values(*) as * range(_time)
as duration by JSESSIONID uri_path</query>
   <earliest>$time_picker.earliest$</earliest>
   <latest>$time_picker.latest$</latest>
</search>
```

Step 2) Use post-process searches to compute # sessions & avg duration by service:

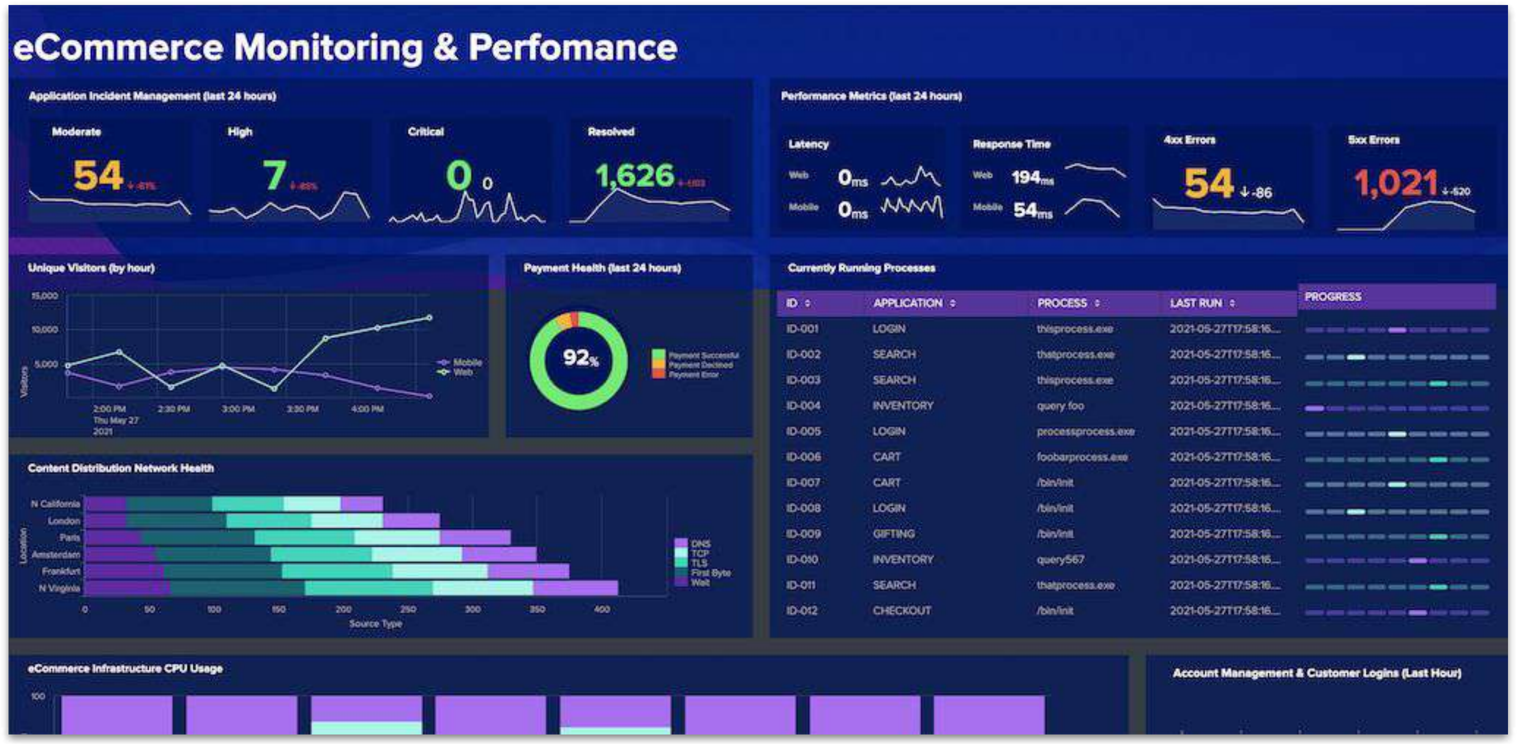
```
<search base="all_web_sessions">
   <query>| stats count as num_sessions by uri_path</query>
</search>

<search base="all_web_sessions">
   <query>| timechart avg(duration) by uri_path</query>
</search>
```

Dashboard Studio

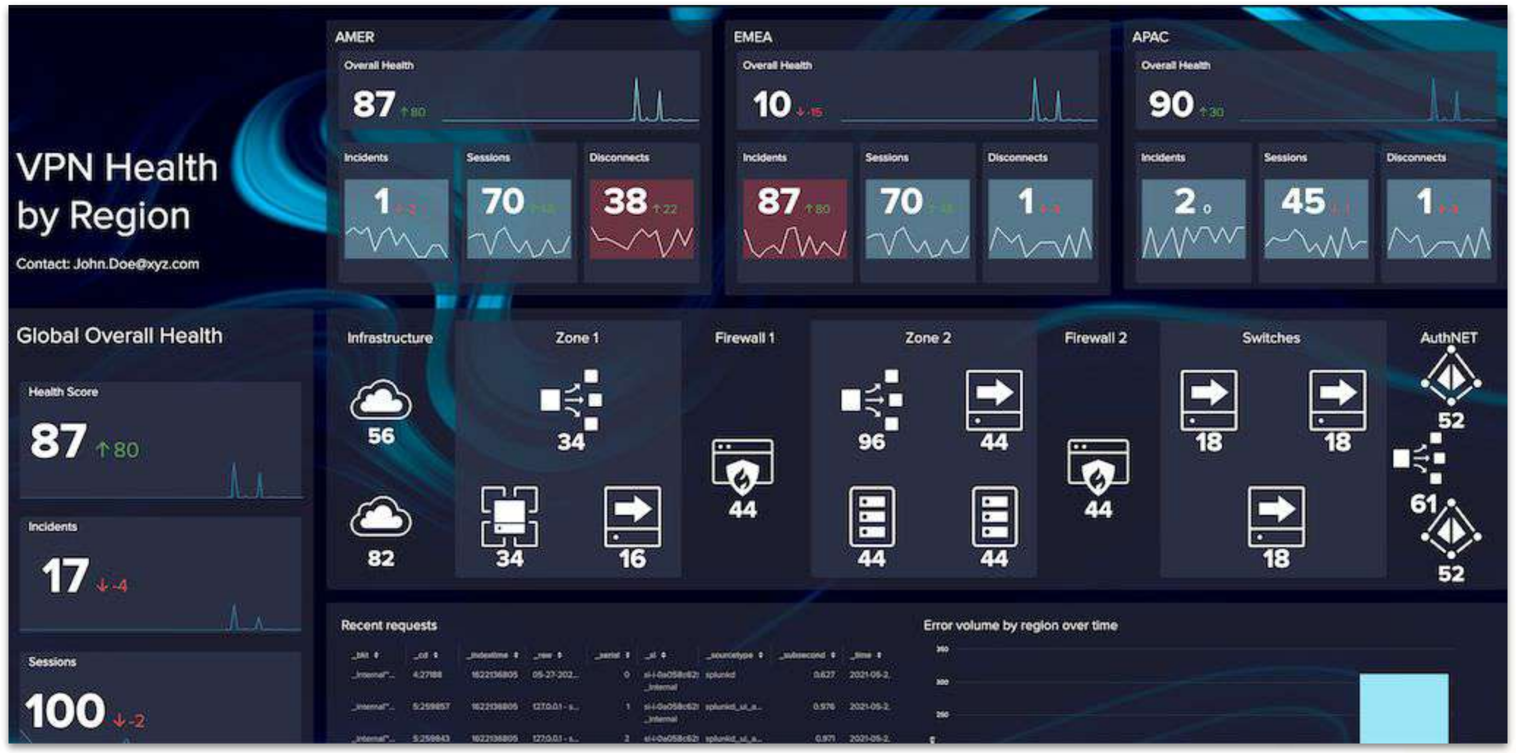
Deliver high-value dashboards & reports to business & operational teams with Dashboard Studio

Example Hub has lots of examples for you to use and customize

eCommerce Monitoring & Performance


eCommerce Monitoring & Perfomance
Application Incident Management (last 24 hours)
Moderate
54
High
7
Critical
0
Resolved
1,626
Performance Metrics (last 24 hours)
Latency
Response Time
4xx Errors
54
5xx Errors
1,021
Unique Visitors (by hour)
Payment Health (last 24 hours)
92%
Currently Running Processes
Content Distribution Network Health
eCommerce Infrastructure CPU Usage
Account Management & Customer Logins (Last Hour)

VPN Health by Region



Geography-Based Analysis

Geography is important because all data takes place in time AND space

Example use cases:

- DDOS attack from Nation State
- Fast-moving logons or badge swipes
- Find out where web-based transactions are coming from – why do we have latency for some transactions?
- Is our sales/marketing strategy in agreement with geography-based segments and outcomes?

Best practices:

- Calibrate geo views to your audience. Show a GOOD map with good info to make decisions
- Don't use "pew pew" laser maps because they're easy, make sure it's valuable
- Use accurate geo info. You can buy updated MaxMind geo database in Splunk, or enrich lat/lon from your own DB

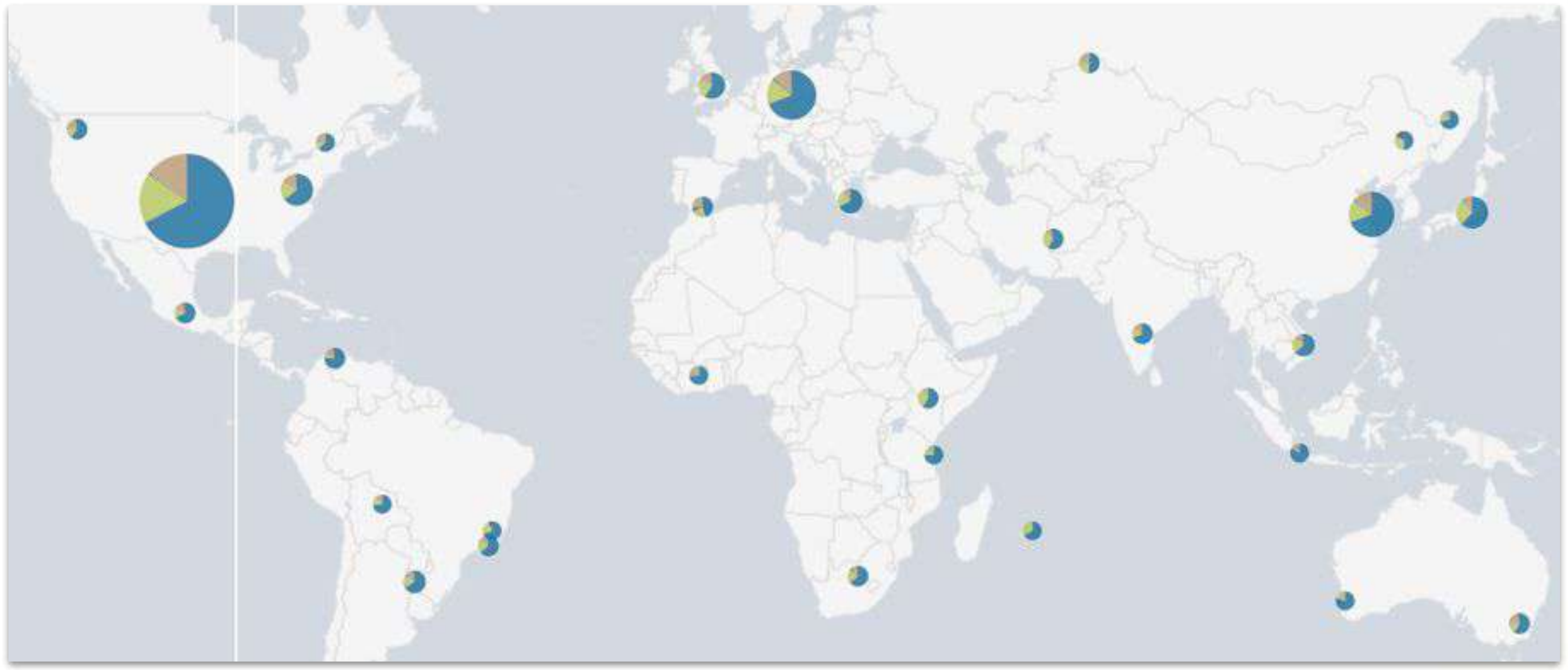
View geographic data with cluster maps

Use cluster maps for localized information (specific latitude/longitude)

Prepare data with iplocation+geostats command

Find all activity by specific locations:

```
sourcetype=access_combined
| iplocation clientip
| geostats count by status
```



Learn more about Cluster Maps:
https://docs.splunk.com/Documentation/Splunk/latest/Viz/MarkerMap

Regex/rex commands

Extract fields at search time with `rex` & `regex` commands. Develop regex and pass back to Splunk admin team.

Best practice: Use a website like regex101.com and Google search to practice regex. E.g., “How do I build regex for: everything up to the first semicolon?”

Example: Splunk admin should already extract timestamp

```
1573704878 200 - User fsamuels failed login
```

```
| rex field="_raw" "(?<status>\d*) \- User (?<username>\S*)"
```

See this example on regex101.com: https://regex101.com/r/BrgKls/1

Regex command:
https://docs.splunk.com/Documentation/Splunk/latest/SearchReference/Regex
Rex command:
https://docs.splunk.com/Documentation/Splunk/latest/SearchReference/Rex

Schema-on-read and Field Extraction

splunk>

Schema-on-read vs Schema-on-write

Splunk has many options for storing and searching data

Schema-on-read / search-time extractions:

- We only write _raw and metadata to disk.
- Extract new fields when you run a search.
- This makes ingest very fast, search very flexible.
- Good for asking questions you didn't know in advance (i.e., during security investigations!!!)

Schema-on-write / index-time extractions:

- We write additional field information to disk at ingest-time (e.g., hash, username, source/dest IPs).
- This adds time to ingest and uses extra disk space, but enables fast, optimized searches.
- Good for asking questions you know you need to ask (i.e., for security monitoring).

Data model acceleration:

- We can write additional field information after the fact.
- Acceleration searches run in background on regular basis.

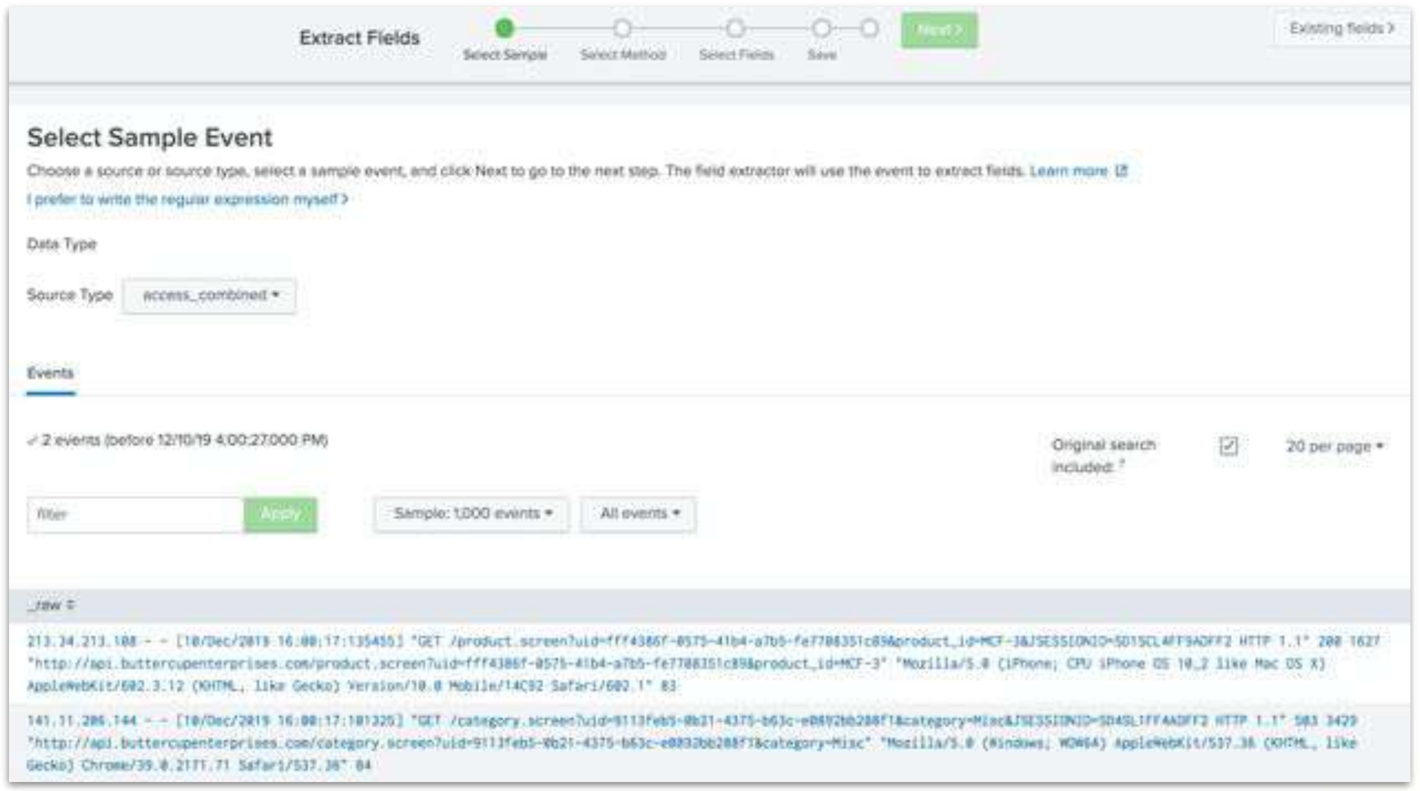
Interactive Field Extractor

Extract fields with UI-based Interactive Field Extractor - it creates regex for you!

Best practice: Extract ALL fields you need using single regex

Bad practice: A customer who used 72 different field extractions for 4 different types of data grouped into one sourcetype.

- **How to fix:** Separate into 4 different sourcetypes, each with 1-3 field extractions covering the ~18 fields on each sourcetype.



Select Sample Event

Choose a source or source type, select a sample event, and click Next to go to the next step. The field extractor will use the event to extract fields. Learn more

I prefer to write the regular expression myself >

Data Type

Source Type: access_combined

Events

✓ 2 events (before 12/10/19 4:00:27.000 PM)

Original search included: ✓ 20 per page

Sample: 1,000 events · All events

```
_raw
213.34.213.108 - - [10/Dec/2019 16:00:17:135455] "GET /product.screen?uid=fff4386f-0575-41b4-a7b5-fe7708351c89&product_id=MCF-3&JSESSIONID=SD1SCL4FF9ADFF2 HTTP 1.1" 200 1627 "http://api.buttercupenterprises.com/product.screen?uid=fff4386f-0575-41b4-a7b5-fe7708351c89&product_id=MCF-3" "Mozilla/5.0 (iPhone; CPU iPhone OS 10_2 like Mac OS X) AppleWebKit/602.3.12 (KHTML, like Gecko) Version/10.0 Mobile/14C92 Safari/602.1" 83
141.11.206.144 - - [10/Dec/2019 16:00:17:101325] "GET /category.screen?uid=9113feb5-0b21-4375-b63c-e0032bb208f1&category=Misc&JSESSIONID=SD4SL1FF4ADFF2 HTTP 1.1" 503 3429 "http://api.buttercupenterprises.com/category.screen?uid=9113feb5-0b21-4375-b63c-e0032bb208f1&category=Misc" "Mozilla/5.0 (Windows; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/39.0.2171.71 Safari/537.36" 84
```

Work with Splunk admin teams on field extractions, but the Interactive Field Extractor is good for a quick one-off

Eventtypes, tags and macros

We can “hide” field extractions under eventtypes and tags. `Eventtype=myfavoritedata =>` `(sourcetype=X type=foo) OR (sourcetype=Y style=bar)`

E.g. tag=network as part of Common Information Model hides LOTS of contributing information to make it easy for you to search for network data

Macro: store reusable search strings. E.g., wrap string “lookup assets.csv host” as macro `` `asset_enrich` ``:

```
sourcetype=X | `asset_enrich`
=> sourcetype=X | lookup assets.csv host
```

Save lots of time if you’re reusing search strings. You can pass variables (`` `mymacro(X)` `` => something)

Best practice: Work with your Splunk admin team

Common Information Model

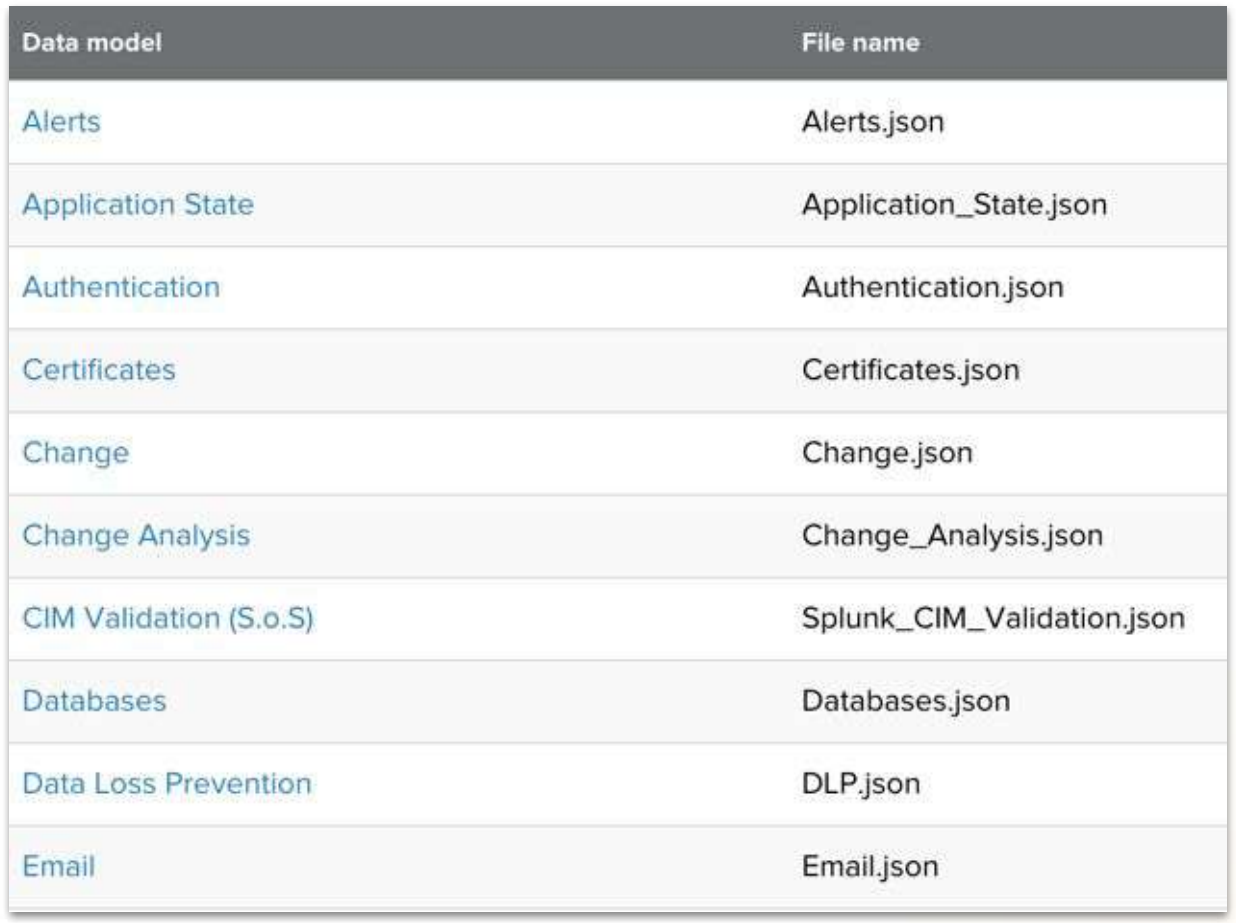
Use field aliases and tags to harmonize data across different types

E.g. field name “src” to appear on all network data, and sometimes represents “source”, “source_ip”, “starting_ip”, etc. This way you can search on src=1.2.3.4 across all data.

Talk to your Splunk admin team. If working with common data types, they’re probably already aligned to CIM. Security teams should definitely align to CIM.

Learn more about the Common Information Model:
https://docs.splunk.com/Documentation/CIM/latest/User/Overview

Validate CIM compliance:
https://github.com/hire-vladimir/SA-cim_vladiator



Data model	File name
Alerts	Alerts.json
Application State	Application_State.json
Authentication	Authentication.json
Certificates	Certificates.json
Change	Change.json
Change Analysis	Change_Analysis.json
CIM Validation (S.o.S)	Splunk_CIM_Validation.json
Databases	Databases.json
Data Loss Prevention	DLP.json
Email	Email.json

Thank you