



# Splunk Search Pro Tips

---

**Dan Aiello, Principal Cyber Security Engineer**

**Splunk .conf2015**

**MITRE**

# Agenda

---

- **My background**
- **Comments**
- **Search by index**
- **In the year 2000**
- **Red Card**
- **Watch Lists**
- **Search Job Inspector**
- **More fun with subsearches**
- **Carrier signal**
- **Imaginary events**
- **Summary**

## My Splunk background

---

- **4 years Splunk experience**
- **SOC is primary user base**
  
- **6 indexers**
- **350 GB data/day**
- **90 indexes**
- **170 sourcetypes**

# comments

---

// No comment

**MITRE**

---

## Comment your Splunk search

---

```
sourcetype=access_combined_wcookie  
| eval COMMENT="This is my comment"
```

or

```
sourcetype=access_combined_wcookie  
| rename COMMENT -> "This is my comment"
```

\* There's nothing special about the word "COMMENT", use whatever you like

## rename vs. eval for comments?

---

- In practice, it does not seem to matter
- In a few odd circumstances, I have seen rename be faster than eval

# Why would you need comments?

source="C:\Network Analysis\stier1rusxwalmartdc\S0-0-0.csv" OR source="C:\Network Analysis\stier1rdinmumbai010-7-1\S2-0.csv" OR source="C:\Network Analysis\stier1rdgbreddit010-1-2\Gig0-2.csv" OR source="C:\Network Analysis\stier2rdingurgao010-5-1\fo-1.csv" OR source="C:\Network Analysis\stier3rdinsecund010-5-2\Gig0-0.csv" OR source="C:\Network Analysis\stier2rdphcebu010-5-1\fo-2-0.csv" host="SEZ00VVM-153" sourcetype="csv" | rex field=source "(?<country>.\*?)\$" | lookup datacentre.csv country OUTPUT receivebandwidth sitename tier | search tier=tier1 | eval Intraffic=IN/1048576 | eval Outtraffic=Out/1048576 | eval result=(Intraffic)+(Outtraffic) | eval seventyperc=receivebandwidth\*0.7 | eval eightyperc=receivebandwidth\*0.8 | eval ninetyperc=receivebandwidth\*0.9 | where result>seventyperc | stats Values(result) AS Inout, values(seventyperc) AS 70%, values(eightyperc) AS 80%, values(ninetyperc) AS 90%, values(receivebandwidth) as 100% count as nc by sitename \_time | bin \_time span=1d | stats sum(nc) as NOC by sitename \_time | eval NOH =NOC\*5/60 | timechart span=1d values(NOH) AS total by sitename

Hi shreyasathavale,

Sure, the lazy and not v

earliest="06/08/2017 00:00:00" | stats count by eventstats max | where hit=maximum | sort hit desc

5. | top limit=1 hit, date\_mday, date\_hour | fields date\_hour, date\_mday, hit, earliest, latest | map search=" search (earliest=\$earliest\$ latest=\$latest\$+3600) index=perfmon host=web1 (counter=\"% Process or Time\" OR counter=\"Get Requests/Sec\" OR counter=\"Current Connections\") | stats avg(Value) by host, counter

This is completely untested and keep in mind, for me it's early Monday morning :)

I'm pretty sure this can be done with some **stats** tricks

Hope that helps ...

cheers, MuS

[Add comment](#)

5. index=my\_index someField="someVALUE" | eval 15daysago=relative\_time(now(), "-15d@d") | eval 14daysago=relative\_time(now(), "-14d@d") | eval date\_created\_tz\_epoch=strptime(date\_created\_tz, "%m/%d/%Y") | search date\_created\_tz\_epoch<15daysago | stats count by severity | rename count as resCOL1 | eval label="Sev - ".severity | addcoltotals | resCOL2 resCOL3 resCOL4 resCOL5 resCOL6

5. index=my\_index someField2="someVALUE2" | eval 15daysago=relative\_time(now(), "-15d@d") | eval 14daysago=relative\_time(now(), "-14d@d") | eval date\_created\_tz\_epoch=strptime(date\_created\_tz, "%m/%d/%Y") | search date\_created\_tz\_epoch<15daysago AND date\_created\_tz<15daysago | stats count by severity | rename count as resCOL2 | eval label="Sev - ".severity | addcoltotals | resCOL3 resCOL4 resCOL5 resCOL6

5. index=my\_index (NOT someField2="someVALUE2") | eval 15daysago=relative\_time(now(), "-15d@d") | eval 14daysago=relative\_time(now(), "-14d@d") | eval date\_created\_tz\_epoch=strptime(date\_created\_tz, "%m/%d/%Y") | search date\_created\_tz\_epoch<15daysago AND date\_created\_tz<15daysago | stats count by severity | rename count as resCOL2 | eval label="Sev - ".severity | addcoltotals | resCOL3 resCOL4 resCOL5 resCOL6

Most Recent Activity:  
Edited by gcat0  
161 • 1 • 1 • 5

Question by deepthi5  
3 days ago  
23 • 1 • 2 • 11

Most Recent Activity:  
Commented by MuS •  
36.7k • 80 • 323 • 907

Question by imanpoelri  
4 days ago  
80 • 2 • 3 • 40

Answer by MuS •  
Jun 21 at 01:34 PM








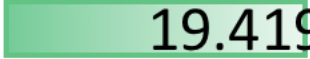
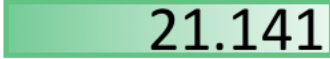
# Search by index

---



## Searching by index and sourcetype

Specifying an index in your search speeds it up

	Fast		Smart		Verbose
index=	 6.27		 6.384		 6.593
index= AND sourcetype=	 8.109		 7.048		 6.447
sourcetype=	 7.07		 19.419		 21.141

This difference is less pronounced in Fast Mode



# In the year 2000

---



## In the year 3000

**MITRE**

---

## Get with the times

---

- **Timestamps are extremely important for Splunk data**
- **Detected at index time, set forever**
- **Cannot be fixed if they're wrong**
  
- **Common errors:**
  - Incorrect time zone interpretation
  - Host clock incorrect

## Past, present, future

If this search ever returns events, you have timestamp problems<sup>1</sup>:

```
index=* earliest=+30m latest=+9y
```

This requires some tweaking, depending on your expected delay:

```
index=* | eval delta=_indextime-_time | where  
delta>300
```

<sup>1</sup> Or a flux capacitor<sup>2</sup>

<sup>2</sup> Or a TARDIS

# red card

---

your approximate wait time is...

**MITRE**

---

## Calculate average delay proxy logs

---

```
index=main  
| eval delta = _indextime - _time  
| timechart span=1h avg(delta)
```

**Problem: that's a *lot* of events**



## Calculate average delay proxy logs

### Solution:

```
*/5 * * * * wget http://testdomain.zzz
```

```
index=main testdomain.zzz  
| eval delta = _indextime - _time  
| timechart span=1h avg(delta)
```

Search terms	Duration
index=main	453 s
index=main testdomain.zzz	6 s



# watchlists

---

**better than grep -f**

**MITRE**

---

## Watchlist examples

---

- **Known “evil” IP addresses**
- **Known “evil” domain names**
- **List of your DMZ web servers**
- **Known allowed IP/port combinations in your DMZ**

# Example IP watchlist

Search

Pivot

Reports

Alerts

Dashboards

Search & Reporting

🔍 New Search

Save As ▾Close

inputlookup ip\_watchlist.csv

All time ▾🔍

✓ 0 events (before 7/17/15 4:07:36.000 AM)

Job ▾⏸■↶⬇🖨🗨 Verbose Mode ▾

Events

Patterns

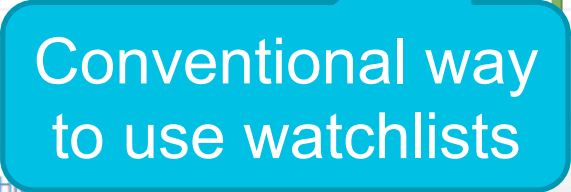
Statistics (22)

Visualization

20 Per Page ▾Format ▾Preview ▾

< Prev12Next >

clientip ↕	type ↕
131.178.233.243	malicious
212.58.253.71	malicious
59.36.99.70	malicious
223.205.219.67	malicious
86.9.190.90	malicious
142.233.200.21	malicious
207.36.232.245	malicious
170.192.178.10	malicious
91.217.178.210	malicious
89.106.20.218	malicious
50.23.124.50	malicious



This is essentially  
grep -F

**New Search** Save As ▾ Close

sourcetype=access\_combined\_wcookie [| inputlookup ip\_watchlist.csv | search type=malicious | fields clientip]

✓ 2,572 events (before 8/4/15 5:12:57.000 PM) Job ▾ || ■ ↓ 🖨 💡 Smart Mode ▾

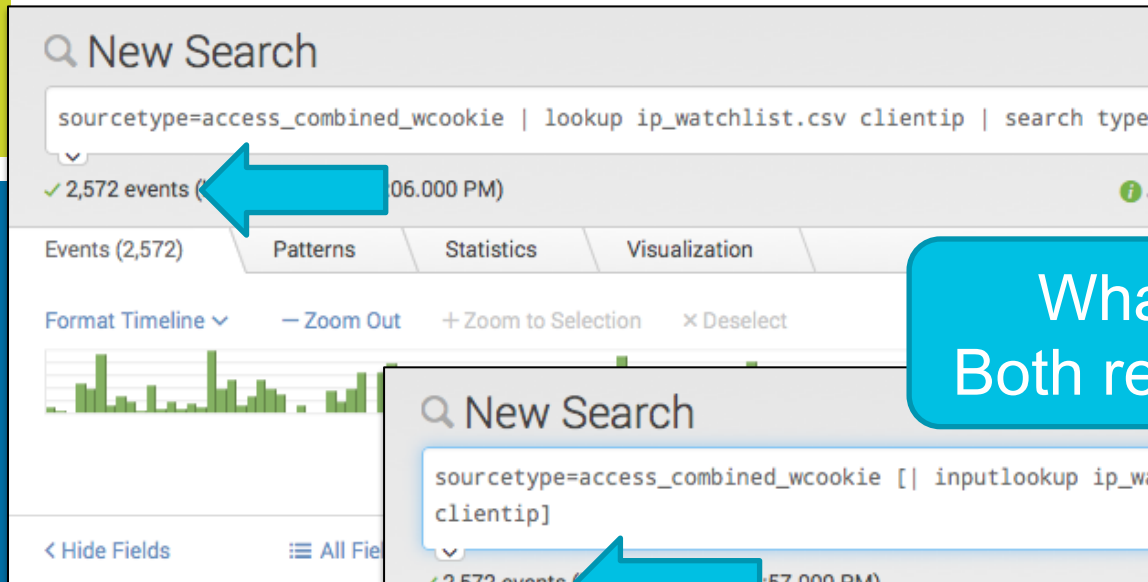
Events (2,572) Patterns Statistics

Format Timeline ▾ — Zoom Out + Zoom to Selection + Preset 1 day per column

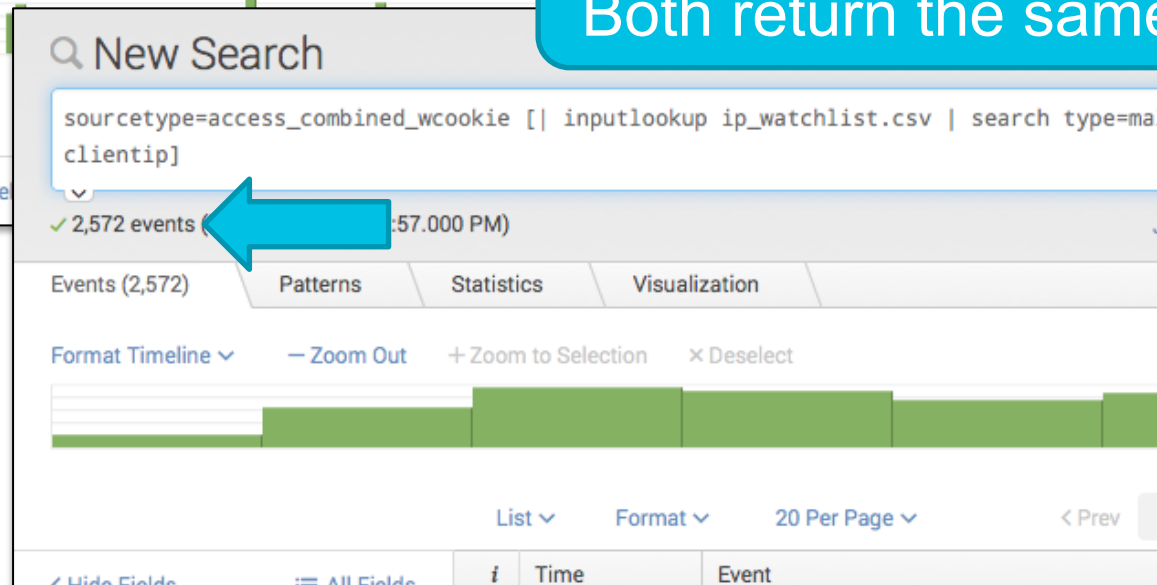
Let's try a subsearch

List ▾ Format ▾ 20 Per Page ▾ < Prev 1 2 3 4 5 6 7 8 9 ... Next >

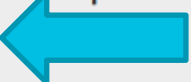
< Hide Fields <span>☰ All Fields</span>		i	Time	Event
<b>Selected Fields</b> <a>host</a> 1 <a>source</a> 3 <a>sourcetype</a> 1		>	7/14/15 5:45:30.000 PM	188.173.152.100 - - [14/Jul/2015:17:45:30] "POST /oldlink?itemId=EST-26&JSESSIONID=SD3SL10FF6ADFF52964 HTTP 1.1" 200 1645 "http://www.buttercupgames.com/oldlink?itemId=EST-26" "Mozilla/5.0 (compatible; MSIE 9.0; Windows NT 6.1; WOW64; Trident/5.0; BOIE9;ENUS)" 306  host = ip-172-31-44-110   source = tutorialdata.zip:/www2/access.log   sourcetype = access_combined_wcookie
<b>Interesting Fields</b> <a>action</a> 5		>	7/14/15 5:45:29.000 PM	188.173.152.100 - - [14/Jul/2015:17:45:29] "GET /category.screen?categoryId=TEE&JSESSIONID=SD3SL10FF6ADFF52964 HTTP 1.1" 200 1226 "http://www.buttercupgames.com/cart.do?action=view&itemId=EST-15&productId=WC-SH-T02" "Mozilla/5.0 (c



What's the difference?  
Both return the same events

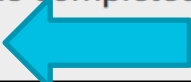


## Search job inspector


This search has completed and has returned **2,572** results by scanning **39,532** events in **1.821** seconds. 

What's the difference?  
*71% less time*


## Search job inspector

This search has completed and has returned **2,572** results by scanning **2,692** events in **0.52** seconds. 

Saving time on a search  
can be important for large  
or frequent searches



This search has completed and has returned 24 results by scanning 6,134,801 events  
124.509 seconds.



This search has completed and has returned 24 results by scanning 266 events in  
6.203 seconds.

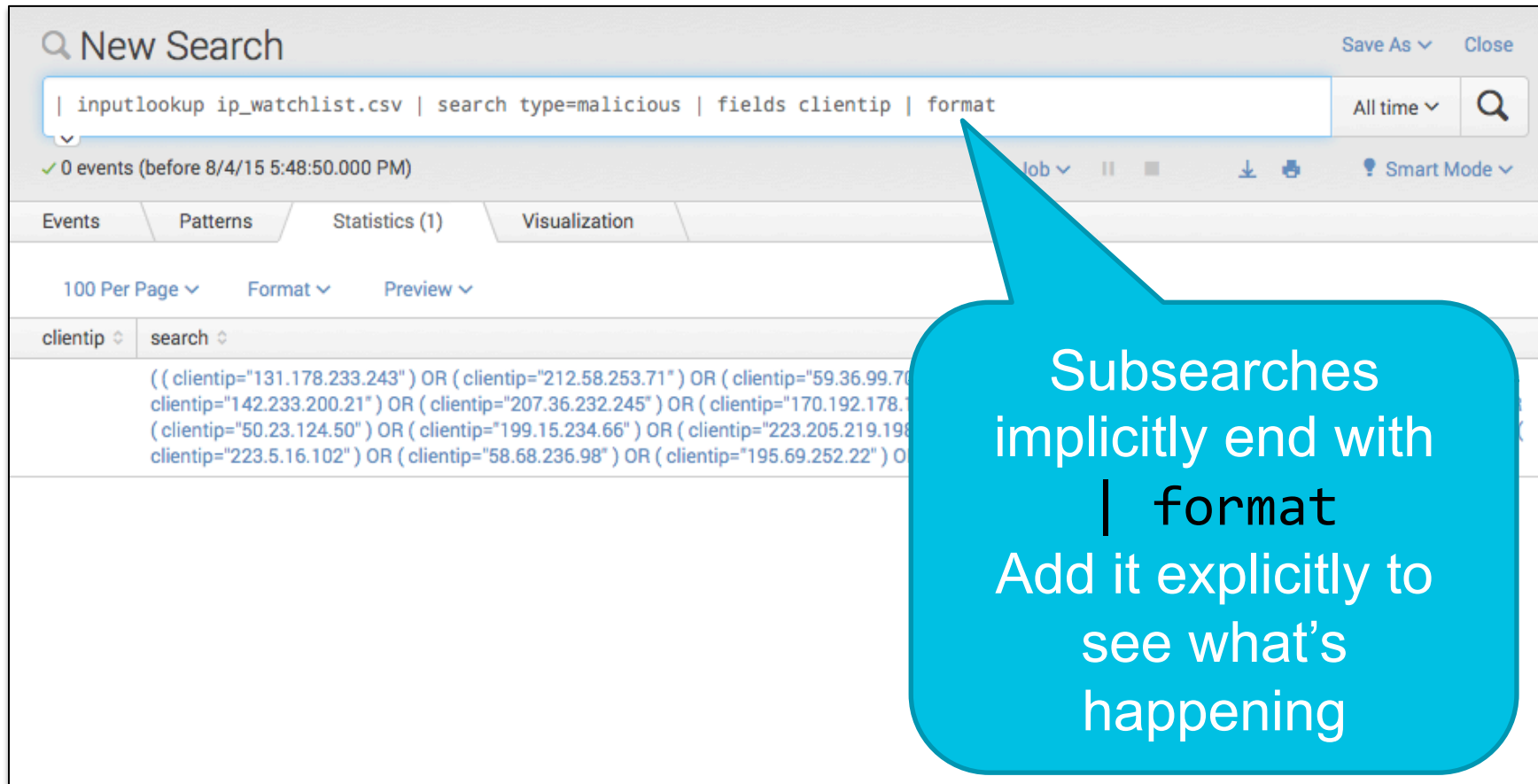
Small watchlists and  
large datasets make  
this difference greater



## How does it work?

The screenshot shows the 'New Search' interface in Splunk. The search bar contains the query: `| inputlookup ip_watchlist.csv | search type=malicious | fields clientip`. Below the search bar, it indicates '0 events (before 8/4/15 5:46:02.000 PM)'. The results are displayed in a table with the following columns: 'clientip'. The table contains 10 rows of IP addresses. A blue callout box with the text 'Just the subsearch' points to the search bar.

clientip
131.178.233.243
212.58.253.71
59.36.99.70
223.205.219.67
86.9.190.90
142.233.200.21
207.36.232.245
170.192.178.10
91.217.178.210
89.106.20.218



The screenshot shows the Splunk Search interface. The search bar contains the query: `| inputlookup ip_watchlist.csv | search type=malicious | fields clientip | format`. Below the search bar, it indicates "0 events (before 8/4/15 5:48:50.000 PM)". The interface includes tabs for Events, Patterns, Statistics (1), and Visualization. A callout box points to the `| format` command in the search bar.

Subsearches  
implicitly end with  
`| format`  
Add it explicitly to  
see what's  
happening

```
sourcetype=access_combined_wcookie  
[ | inputlookup ip_watchlist.csv | search  
type=malicious | fields clientip ]
```

...after the subsearch is evaluated becomes this:

```
sourcetype=access_combined_wcookie  
( ( clientip="131.178.233.243" ) OR  
( clientip="212.58.253.71" ) OR ... )
```

i.e., the results of the subsearch are appended

**normalizedSearch**

```
litsearch sourcetype=access_combined_wcookie ( ( clientip="131.178.233.243" )  
OR ( clientip="212.58.253.71" ) OR ( clientip="59.36.99.70" ) OR ( clientip="223.205.219.67" ) OR ( clientip="86.9.190.90" ) OR ( clientip="142.233.200.21" ) OR ( clientip="207.36.232.245" ) OR ( clientip="170.192.178.10" ) OR ( clientip="91.217.178.210" ) OR ( clientip="89.106.20.218" ) OR ( clientip="50.23.124.50" ) OR ( clientip="199.15.234.66" ) OR ( clientip="223.205.219.198" ) OR ( clientip="203.45.206.135" ) OR ( clientip="212.27.63.151" ) OR ( clientip="223.5.16.102" ) OR ( clientip="58.68.236.98" ) OR ( clientip="195.69.252.22" ) OR ( clientip="69.80.0.18" ) OR ( clientip="188.173.152.100" ) ) | fields keepcolorder=t "*" "_bkt" "_cd" "_si" "host" "index" "linecount" "source" "sourcetype" "splunk_server"
```

The Search Job Inspector shows us this.

**Why is**

```
sourcetype=access_combined_wcookie  
( ( clientip="131.178.233.243" ) OR  
( clientip="212.58.253.71" ) OR ... )
```

**Better than**

```
sourcetype=access_combined_wcookie  
| lookup ip_watchlist.csv clientip | search  
type=malicious
```

# Search Job Inspector

---

...explains it all

**MITRE**

---

This icon means there's some debugging message you should examine

The screenshot shows the MITRE ATT&CK framework interface. At the top, there's a search bar with the query "sourcetype=access\_combined\_wcookie | lookup ip\_watchlist.csv | search type=malicious". Below the search bar, it indicates "2,572 events (before 8/22/15 10:00:17.000 PM)". A green icon with an 'i' inside a circle is visible next to the "Job" label. A context menu is open over this icon, showing options: "Assuming implicit lookup table with filename 'ip\_watchlist.csv'", "Edit Job Settings", "Send Job to Background", "Inspect Job", and "Delete Job". The "Inspect Job" option is highlighted. The interface also shows a timeline visualization with green bars representing events over time.

*Inspect Job*  
is always here

## Search job inspector

This search has completed and has returned **2,572** results by scanning **39,532** events in **1.974** seconds.

The following messages were returned by the search subsystem:

INFO: Assuming implicit lookup table with filename 'ip\_watchlist.csv'.

(SID: 1440247286.13) [search.log](#)

### Execution costs

Duration (seconds)	Component	Invocations	Input count	Output count
	0.02 command.fields	21	2,572	2,572
█	0.06 command.lookup	21	39,532	39,532
██████████	1.57 command.search	42	39,532	42,104
█	0.21 command.search.filter	39	-	-
	0.03 command.search.index	21	-	-
	0.02 command.search.calcfields	18	39,532	39,532
	0.02 command.search.fieldalias	18	39,532	39,532
	0.00 command.search.index.usec_1_8	246	-	-
	0.00 command.search.index.usec_8_64	20	-	-
█	0.47 command.search.rawdata	18	-	-
█	0.46 command.search.kv	18	-	-
█	0.40 command.search.typer	18	39,532	39,532
	0.02 command.search.lookups	18	39,532	39,532
	0.02 command.search.tags	18	39,532	39,532
	0.01 command.search.summary	21	-	-
	0.00 dispatch.check_disk_usage	1	-	-
	0.02 dispatch.createdSearchResultInfrastructure	1	-	-
█	0.04 dispatch.evaluate	1	-	-
█	0.04 dispatch.evaluate.search	2	-	-
	0.00 dispatch.evaluate.lookup	1	-	-
██████████	1.28 dispatch.fetch	22	-	-
██████████	1.66 dispatch.localSearch	1	-	-
	0.00 dispatch.preview	1	-	-



## Search job inspector

| 33 |

This search has completed and has returned 2,572 results by scanning 39,532 events in 1.974 seconds.

The following messages were returned by the search subsystem:

INFO: Assuming implicit lookup table with filename 'ip\_watchlist.csv'.

(SID: 1440247286.13) [search.log](#)


Profiling information

Debugging message


### Execution costs

Duration (seconds)		Component			
	0.02	command.fields	21	2,572	2,572
█	0.06	command.lookup	21	39,532	39,532
██████████	1.57	command.search	42	39,532	42,104
█	0.21	command.search.filter	39	-	-
	0.03	command.search.index	21	-	-
	0.02	command.search.calcfields	18	39,532	39,532
	0.02	command.search.fieldalias	18	39,532	39,532
	0.00	command.search.index.usec_1_8	246	-	-
	0.00	command.search.index.usec_8_64	20	-	-
██████████	0.47	command.search.execute	18	-	-

## Search job inspector

This search has completed and has returned **2,572** results by  **39,532** events in **1.821** seconds.

## Search job inspector

This search has completed and has returned **2,572** results by  **2,692** events in **0.52** seconds.

The slowest parts of a Splunk search are usually field extraction and reading events from disk.

## Approximate order of operations for searches

---

- 1. Search index for keywords**
- 2. Read matching events from disk**
- 3. Extract fields (as necessary)**
- 4. Match keywords to fields (as necessary)**
- 5. Filter (e.g. additional “where” or “search” pipes)**
- 6. Send data to search head**

## What are keywords?

**keywords**`sourcetype::access_combined_wcookie type::malicious`**keywords**`clientip::131.178.233.243 clientip::142.233.200.21 clientip::170.192.178.10  
clientip::188.173.152.100 clientip::195.69.252.22 clientip::199.15.234.66  
clientip::203.45.206.135 clientip::207.36.232.245 clientip::212.27.63.151  
clientip::212.58.253.71 clientip::223.205.219.198 clientip::223.205.219.67  
clientip::223.5.16.102 clientip::50.23.124.50 clientip::58.68.236.98  
clientip::59.36.99.70 clientip::69.80.0.18 clientip::86.9.190.90  
clientip::89.106.20.218 clientip::91.217.178.210  
sourcetype::access_combined_wcookie`

## A stitch in time saves nine

lookup	subsearch	
2	21	Check index for keywords
39,000	2,700	Read matching events from disk
39,000	2,700	Extract fields (i.e. regex)
39,000	2,700	Match keywords to fields
39,000	2,700	Filter

\* This is illustrative and approximate, not precise

Pare your data early to save time late















lookup method reads  
and regexes all this data

sourcetype=access\_  
combined\_wcookie

(( clientip="131.178.2  
33.243" ) OR  
( clientip="212.58.253.  
71" ) OR ... )

subsearch method  
reads only this data

## Compare “lookup” and “subsearch” methods

	lookup			subsearch	
Component	Duration	Input count		Duration	Input count
command.search	 1.57	 39,532		 0.17	-
command.search.filter	 0.21	-		0.00	-
command.search.index	 0.03	-		 0.02	-
command.search.rawdata	 0.47	-		 0.08	-
command.search.kv	 0.46	-		 0.04	-
command.search.typer	 0.40	 39,532		 0.03	 2,572

# More fun with subsearches

---



## Field name mismatch with subsearch

---

**For lookup and subsearch, sometimes fields need to be renamed**

**lookup method**

```
| lookup watchlist.csv foo AS bar
```

**subsearch method**

```
[ | inputlookup watchlist.csv  
  | rename foo AS bar ]
```

The screenshot shows the Splunk Search & Reporting interface. The top navigation bar includes 'splunk', 'App: Search & Reporting', and various user and system links. Below this is a green bar with 'Search', 'Pivot', 'Reports', 'Alerts', and 'Dashboards'. The main area is titled 'New Search' and contains a search bar with the query: `|inputlookup ip_watchlist.csv | search type=malicious | fields clientip | rename clientip -> query | format`. The search results show '0 events (before 7/22/15 4:19:43.000 AM)'. A blue callout bubble points to the 'query' field in the search bar.

query	search
(("131.178.233.243") OR ("212.58.253.71") OR ("59.36.99.70") OR ("223.205.219.67") OR ("86.9.190.90") OR ("170.192.178.10") OR ("91.217.178.210") OR ("89.106.20.218") OR ("50.23.124.50") OR ("199.15.234.66") OR ("212.27.63.151") OR ("223.5.16.102") OR ("58.68.236.98") OR ("207.36.232.245") OR ("203.45.206.135") OR ("	

If you rename a field to “query”, you can search anywhere in the event rather than a single field

# Large subsearches

The screenshot shows the Splunk Search interface. At the top, the search bar contains the query: `sourcetype=access_combined_wcookie [inputlookup alexa-top-44k.csv | sort limit=0 -rank | fields domain | rename domain -> query ]`. Below the search bar, it indicates "0 events (7/5/15 12:00:00.000 AM to 8/4/15 2:13:48.000 AM)". The interface includes tabs for Events (0), Patterns, Statistics, and Visualization. A message at the bottom left states "No results found." A context menu is open over the "Job" button, displaying the error: "[subsearch]: Subsearch produced 44000 results, truncating to maxout 10000." The menu also includes options: "Edit Job Settings", "Send Job to Background", "Inspect Job", and "Delete Job". A blue callout bubble points to the error message.

If your watchlist is >10000 lines, the subsearch method chokes

# Large subsearches

New Search

Save As Close

sourcetype=access\_combined\_wcookie [inputlookup alexa-top-44k.csv | sort limit=0 -rank | fields domain | rename domain -> query | format ]

Last 30 days

4,017 events (7/5/15 12:00:00.000 AM - 7/15 2:17:15.000 AM)

Job

Smart Mode

Events (4,017)

Patterns

Format Timeline

Zoom

1 day per column

Edit Job Settings

Send Job to Background

Inspect Job

Delete Job

Warning is gone

List Format 20 Per Page

< Prev 1 2

< Hide Fields All Fields

i	Time	Event
	7/14/15 6:22:15.000 PM	91.205.189.15 - - [14/Jul/2015:18:22:15] "GET /category.screen?categoryId=SH0TER&JSESSIONID=SD6SL7FF7ADFF53113 HTTP 1.1" 200 1369 "http://www.google.com" "Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/536.5 (KHTML, like Gecko) Chrome/19.0.1084.46 Safari/536.5" 779 host = ip-172-31-44-110 source = tutorialdata.zip:/www2/access.log

We have events

Add "| format" explicitly to fix it!

# carrier signal

---

## What's the problem with watchlists?

---

**When they don't alert, is it because:**

- **the watchlist is broken?**
- **there's nothing to alert on?**

# Add test cases to your watchlist

The screenshot shows the Splunk Search & Reporting interface. The search bar contains the query `|inputlookup referer_watchlist.csv`. Below the search bar, the results are displayed in a table. A blue callout bubble points to the `type` field in the table, containing the text: "Label your test domain as `type=test` in the watchlist".

referer_domain	type
http://www.bing.com	malicious
http://testdomain.zzz	test

The screenshot shows the Splunk Search & Reporting interface with a search query in the search bar: `sourcetype="access_combined_wcookie" [| inputlookup referer_watchlist.csv | search type=test | fields referer_domain]`. A blue callout bubble points to the subsearch `| search type=test` with the text: "And adjust your subsearch accordingly".

# Add test cases to your watchlist

The screenshot shows the Splunk Search & Reporting interface. The search bar contains the query `|inputlookup referer_watchlist.csv`. Below the search bar, the results are displayed in a table with columns for Events and Patterns. A filter for `type` is applied, showing results for `malicious` and `test`. A blue speech bubble points to the `type=test` filter, and an orange speech bubble points to the search bar.

Label your test domain as `type=test` in the watchlist

What's so great about that?  
We could have just used `google.com` for that test

And adjust your subsearch accordingly



splunk> App: Search & Reporting admin Messages Settings Activity Help

### Save As Alert

Title

Description

Alert type ☒ Scheduled ☐ Real Time

Time Range

Schedule At  minutes past the

Trigger condition

Trigger if number of results

With your wget, you know *precisely* how many to expect and you can alert only when it's erroneous

© 2015 The MITRE Corporation. All rights reserved. MITRE

# imaginary events

---

we landed on the moon

**MITRE**

---

## Creating data on the fly

The screenshot displays the Splunk Search & Reporting interface. At the top, the navigation bar includes the Splunk logo, the current app 'App: Search & Reporting', and various user and system links like 'admin', 'Messages', 'Settings', 'Activity', 'Help', and a 'Find' search bar. Below this, a green header bar contains the main navigation tabs: 'Search', 'Pivot', 'Reports', 'Alerts', and 'Dashboards'. The 'Search' tab is active, and the page title is 'Search & Reporting'.

The main content area is titled 'New Search'. It features a search input field containing the query '| stats count'. To the right of the input field are 'Save As' and 'Close' buttons. Below the input field, a status bar indicates '0 events (before 7/22/15 5:41:58.000 AM)'. To the right of this status bar are buttons for 'Job', a pause icon, a stop icon, a download icon, a print icon, and a 'Smart Mode' toggle.

Below the status bar, there are four tabs: 'Events', 'Patterns', 'Statistics (1)', and 'Visualization'. The 'Statistics (1)' tab is selected. Under this tab, there are three sub-tabs: '20 Per Page', 'Format', and 'Preview'. The 'Format' sub-tab is selected, showing a table with one column labeled 'count' and one row with the value '0'.

## How is this helpful?

You can add to a watchlist inline

**New Search** Save As ▾ Close

`| inputlookup referer_watchlist.csv  
| append [ |stats count | eval referer_domain="http://testdomain.zzz" | eval type="test" ]` All time ▾ Q

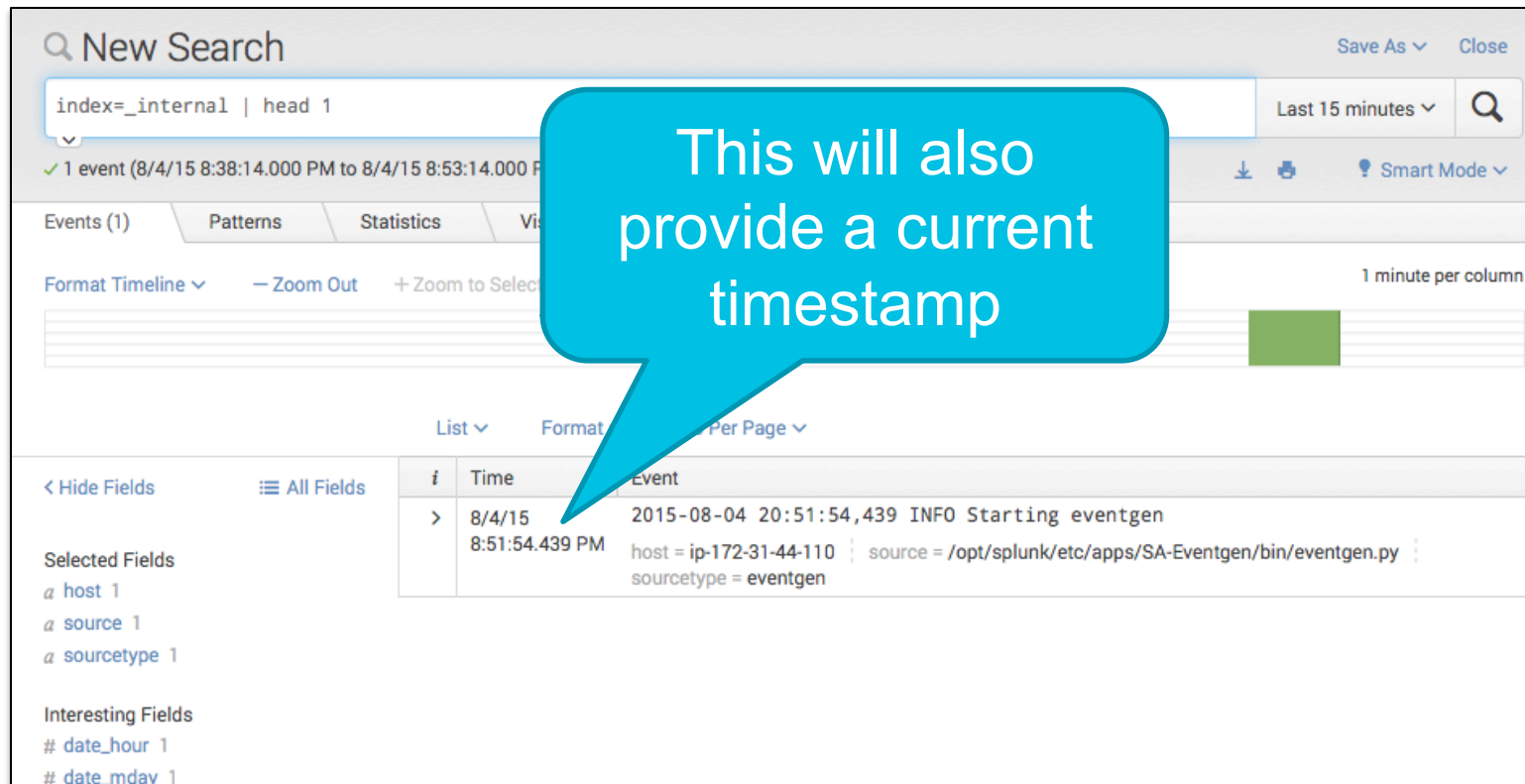
✓ 0 events (before 8/26/15 2:29:31.000 PM) Job ▾ ▮ ▴ ▾ Smart Mode ▾

**Events** **Patterns** **Statistics (3)** **Visualization**

100 Per Page ▾ Format ▾ Preview ▾

count ▾	referer_domain ▾	type ▾
	http://www.bing.com	malicious
	http://www.buttercupgames.com	malicious
0	http://testdomain.zzz	test

## Creating data on the fly with timestamps



New Search

index=\_internal | head 1

1 event (8/4/15 8:38:14.000 PM to 8/4/15 8:53:14.000 PM)

Events (1) Patterns Statistics Visualize

Format Timeline Zoom Out Zoom to Selection

1 minute per column

List Format Per Page

i	Time	Event
>	8/4/15 8:51:54.439 PM	2015-08-04 20:51:54,439 INFO Starting eventgen host = ip-172-31-44-110 source = /opt/splunk/etc/apps/SA-Eventgen/bin/eventgen.py sourcetype = eventgen

< Hide Fields All Fields

Selected Fields

- a host 1
- a source 1
- a sourcetype 1

Interesting Fields

- # date\_hour 1
- # date\_mday 1

This will also provide a current timestamp

# Are we there yet?

---

## Overall lessons

---

- **Read *Splunk Search Manual***
- **Try multiple methods**
- **Use the Job Inspector**
- **Understand what Splunk is doing “under the hood”**



# Thank you!

---

**MITRE**

---