

Metadata

_sourceHost	The host name of the Source. For local Sources the name of the Source is set when you configure the Source. For remote Collectors, this field uses the remote host's name. The _sourceHost metadata field is populated using a reverse DNS lookup. If the name cannot be resolved, _sourceHost is displayed as localhost.	<code>_source eHost = *MySQL*</code>
_sourceName	The name of the log file, determined by the path you entered when you configured the Source.	<code>_sourceName=/path/to/file/</code> <code>_sourceName=*path*</code> <code>_sourceName = " /var/log/tomcat/logs/foobar.log"</code>
_sourceCategory	This field is created when you enter text into the Source Category field at Source configuration time. Log categories can be somewhat complex, as many log files may belong to more than one logical category.	<code>_sourceCategory=OS*</code> <code>_sourceCategory=*Application*</code> <code>_sourceCategory=*Audit</code>
_collector	Returns results from the named Collector only. Entered when a Collector is installed and activated.	<code>_collector=publ ic_ cloud</code>
_source	Returns results from the named Source only. Entered when a Source is configured.	<code>_source=* syslog*</code>

While `_sourceName = *api.log works`, `_sourceName = " *api.log"` will fail.

List all categories: `* | count by _sourceCategory | fields -_count`

Input format

keyvalue	For KVP	<code> keyvalue "age"</code>
	type logs.	<code> keyvalue infer "hairColor", "lastVisit"</code>
	The	<code> keyvalue regex "(.*)[,]" keys "serviceinfo.IP", "loggingcontext.region", "request.m</code>
	keyvalue	<code> keyvalue auto</code>
	operator	
	allows	
	you to get	
	values	
	from a log	
	message	
	by	
	specifying	
	the key	
	paired	
	with each	
	value.	



Input format (cont)

csv The csv operator allows you to parse Comma Separated Values (CSV) formatted log entries. It uses a comma as the default delimiter.

Parse comma delimited fields
`| csv_raw extract 1 as user, 2 as id, 3 as name`

Parse a stream query and extract search terms
`" Starting stream query" | parse " query=[*], queryI d" as query | csv query extract search`

Specify an escape, and quote character
`csv fieldName escape='\', quote='' extract A, B, _, _, E, F`

JSON The JSON operator allows you to extract values from JSON input. Because JSON supports both nested keys and arrays that contain ordered nested keys and keys in arrays.

Extracting a single top-level field
`_sourc eCa teg ory =stream RawOut put Pro cessor " \"me ssa ge \" \" | parse " exp lai nJs or -jso nobje ct`

Extracting multiple fields
`_sourc eCa teg ory =stream RawOut put Pro cessor " \"me ssa ge \" \" | parse " exp lai nJs or erI d" | fields -jso nobje ct`

Extracting a nested key
`* | json field= jso nobje ct " met a.t ype "`

Finding values in a JSON array
`* | json field= jso nobje ct " bas eli neI nte rva ls"`

Refer to one specific entry in an array
`* | json field= jso nobje ct " bas eli neI nte rva ls[ 1]"`

Using the nodrop option
`* | json field= jso nobje ct " bas eli neI nte rva ls[ 0]" nodrop`

Note: The JSON operator also supports the nodrop option, which allows messages containing invalid JSON values to be displayed.



Input format (cont)

Using wildcard (*)

```
_sourc eCa teg ory =0365* | json " Act or[ *].T yp e" as
Actortype
```

json auto works by searching for json blobs beginning at the end of the message. Usually logs begin with a preamble, such as a timestamp. In cases where content appears at the end of the message after the json blob, the extraction could fail. Having the json blob at the end of the message is recommended, as having it in the middle could cause extraction failure.

KVP: Key-Value Pairs. Logs formatted this way look something like this:

```
[2019- 12-24 23:59: 59.380 +1100] age=42 name="Rick Deckar d" hairCo lor ="br own " lastVi sit ="20 18- 04
-19 13:00"
```

infer: Default mode. Uses an internal list of regex to extract the value for a key.

regex: In Regular Expression mode, you must explicitly match keys and values based on a regex.

auto: Extract up to N fields. N is 100 by default.

Conditions

if There are two forms of ternary expression you can use in Sumo Logic queries: one is constructed using the IF operator, and the other uses the question mark (?) operator. These expressions are used to evaluate a condition as either true or false, with values assigned for each outcome. It is a shorthand way to express an if-else condition.

```
| if(status_code matches " 5*", 1, 0) as server_error
```

```
| status_code matches " 5*" ? 1 : 0 as server_error
```

in The In operator returns a Boolean value: true if the specified property is in the specified object, or false if it is not.

```
| if (status_code in ("500", " 501 ", " 502 ", " 503 ", " 504 ", " 505 ", " 506 ", " 401")) as status_code_type
```

where The where operator must appear as a separate operator distinct from other operators, delimited by the pipe symbol ("|").

```
//We recommend placing inclusive filters before exclusive filters in query strings
```

```
| where status _code matches " 4*"
```

```
| where !(stat us_code matches " 2*")
```



By **TME520** (TME520)
cheatography.com/tme520/tme520.com

Published 4th September, 2017.
Last updated 26th April, 2020.
Page 3 of 14.

Sponsored by **CrosswordCheats.com**
Learn to solve cryptic crosswords!
<http://crosswordcheats.com>

Conditions (cont)

isBlank	The isBlank operator checks to see that a string contains text. Specifically, it checks to see if a character sequence is whitespace, empty ("") ,or null. It takes a single parameter and returns a Boolean value: true if the variable is indeed blank, or false if the variable contains a value other than whitespace, empty, or null.	where isBlank(user)
isEmpty	The isEmpty operator checks to see that a string contains text. Specifically, it checks to see whether a character sequence is empty ("") or null. It takes a single parameter and return a Boolean value: true if the variable is indeed empty, or false if the variable contains a value other than empty or null.	if(isEmpty(src_ip) ,1,0) as null_counts
isNull	The isNull operator takes a single parameter and returns a Boolean value: True if the variable is indeed null, or false if the variable contains a value other than null.	where isNull(src_ip)

Data extraction

parse (-regex)	Best for variable patterns. Also called the extract operator; enables users to extract more complex data from log lines using regular expressions. Can be used to extract nested fields.	parse "Content=" as content
	Parsing an IP address	parse regex "(?<ip_address> \d{1,3} \. \d{1,3} \. \d{1,3} } \. \d{1,3}) "
	Indicating an OR condition to use non-capturing groups	parse regex "list 101 (accepted denied) (?<protocol>.*?) "



By **TME520** (TME520)
cheatography.com/tme520/
tme520.com

Published 4th September, 2017.
 Last updated 26th April, 2020.
 Page 4 of 14.

Sponsored by **CrosswordCheats.com**
 Learn to solve cryptic crosswords!
<http://crosswordcheats.com>

Data extraction (cont)

parse(anchor) Best for predictable patterns. Also called parse anchor, parses strings according to specified start and stop anchors and labels them as fields for use in subsequent aggregation functions in the query such as sorting, grouping...

split The split operator allows you to split strings into multiple strings, and parse delimited log entries, such as space-delimited formats.

```
_sourceCategory=access_logs | parse "]" " as log_level, text | split text delim=':' esult
```

fields The fields operator allows you to choose which fields are displayed in the results of a query.

```
_sourceCategory=access_logs | parse using public/apache | fields method, status_code
```

limit	The limit operator reduces the number of raw messages or aggregate results returned.	count by _source eCa tegory sort by _count limit 5															
matches	The matches operator can be used to match a string to a pattern.	if (agent matches " *MS IE* " ,"In ternet Explor er", " Oth er") as Browser if (agent matches " *Fi ref ox* " ,"Fi ref ox", Bro wser) as Browser															
timeslice	The timeslice operator segregates data by time period.	timeslice 1h count by _timeslice															
		_sourc ena me= *to mcat* timeslice by 5m count by _timeslice															
	Output of last example:	<table> <thead> <tr> <th>#</th><th>Time</th><th>_count</th></tr> </thead> <tbody> <tr> <td>1</td><td>09/07/2017 11:25:00 AM +1000</td><td>9,234</td></tr> <tr> <td>2</td><td>09/07/2017 11:30:00 AM +1000</td><td>14,496</td></tr> <tr> <td>3</td><td>09/07/2017 11:35:00 AM +1000</td><td>15,988</td></tr> <tr> <td>4</td><td>09/07/2017 11:40:00 AM +1000</td><td>3,383</td></tr> </tbody> </table>	#	Time	_count	1	09/07/2017 11:25:00 AM +1000	9,234	2	09/07/2017 11:30:00 AM +1000	14,496	3	09/07/2017 11:35:00 AM +1000	15,988	4	09/07/2017 11:40:00 AM +1000	3,383
#	Time	_count															
1	09/07/2017 11:25:00 AM +1000	9,234															
2	09/07/2017 11:30:00 AM +1000	14,496															
3	09/07/2017 11:35:00 AM +1000	15,988															
4	09/07/2017 11:40:00 AM +1000	3,383															



By **TME520** (TME520)
cheatography.com/tme520/tme520.com

Published 4th September, 2017.
 Last updated 26th April, 2020.
 Page 5 of 14.

Sponsored by **CrosswordCheats.com**
 Learn to solve cryptic crosswords!
<http://crosswordcheats.com>

Data extraction (cont)

trace A trace operator acts as a highly sophisticated filter to connect the dots across different log messages. You can use any identifying value with a trace operator (such as a user ID, IP address, session ID, etc.) to retrieve a comprehensive set of activity associated to that original ID.

```
| trace "ID=( [0-9a- fA-F] {4} )" " 7F9 - 2"
```

About limit: Can be used in Dashboard Panels, but in the search they must be included after the first group-by phrase.

About timeslice: Timeslices greater than 1 day cannot be used in Dashboard Live mode.

About trace: Not supported in Live Dashboards or any continuous query.

Crunch numbers

count	Used in	<code> count by url</code>
count_distinct	conjun-	<code> count_distinct(referrer) by status_code</code>
count_frequent	cation with	<code>_sourcename=*tomcat* count_distinct(_sourceName) group by _sourceHost sort by _co</code>

the group operator and a field name. Only the word by is required. The count function is also an operator in its own right and therefore can be used with or without the word by.

sum Sum adds the values of the numerical field being evaluated within the time range analyzed.

```
| sum(by tes _re ceived) group by _sourc eHost
```

avg The | avg(re que st_ rec eived) by _timeslice
averaging
function
(avg)
calculates
the
average
value of
the
numerical
field
being
evaluated
within the
time
range
analyzed.

median In order | parse " val ue= *" as value | pct(value, 50) as median
to
calculate
the
median
value for
a
particular
field, you
can utilize
the
Percentile
(pct)
operator
with a
percentile
argument
of 50.



By **TME520** (TME520)
cheatography.com/tme520/tme520.com

Published 4th September, 2017.
Last updated 26th April, 2020.
Page 6 of 14.

Sponsored by **CrosswordCheats.com**
Learn to solve cryptic crosswords!
<http://crosswordcheats.com>

Crunch numbers (cont)

outlier Given a `_sourceCategory = II S/A ccess | parse regex " \d+ -\d+ -\d+ \d+:\d+:\d+ (?<server _`
series of `hr esh old =3, con sec uti ve= 2,d ire cti on=+-`
time-s-
tamped
numerical
values,
using the
outlier
operator in
a query
can identify
values in a
sequence
that seem
unexpe-
cted, and
would
identify an
alert or
violation,
for
example,
for a
scheduled
search.

sort The sort `| count as page_hits by _sourceHost | sort by page_hits asc`
operator
orders
aggregated
search
results.
The default
sort order
is descen-
ding.

top Use the top `| top 5 _sourceCategory`
operator
with the
sort
operator, to
reduce the
number of
sorted
results
returned.

min The `| min(1, 2) as v`
minimum `// v = 1`
function
returns the
smaller of
two values.

max The `| max(1, 2) as v`
maximum `// v = 2`
function
returns the
larger of
two values.

About count_frequent: You can use the count_frequent operator in Dashboard queries, but the number of results returned is limited to the top 100 most frequent results.

About top: Can be used in Dashboard Panels, but in the search they must be included after the first group-by phrase.

Geo lookup

Sumo `| parse " remote_ip =*" as remote_ip | lookup latitude, longitude, country_code, country_name`
Logic can `ort _count`
match an
extracted
IP
address to
it's
geogra-
phical
location
on a map.
To create
the map,
after
parsing
the IP
addresses
from log
files, the
lookup
operator
matches
extracted
IP
addresses
to the
physical
location
where the
addresses
originated.

logcompare

The logcompare operator allows you to compare two sets of logs: baseline (historical) and target (current). To run a LogCompare operation, you can use the LogCompare button on the Messages tab to generate a properly formatted query `| logcompare timeshift -24h`

About logcompare: Not supported in Dashboards.



By **TME520** (TME520)
cheatography.com/tme520/tme520.com

Published 4th September, 2017.
Last updated 26th April, 2020.
Page 7 of 14.

Sponsored by **CrosswordCheats.com**
Learn to solve cryptic crosswords!
<http://crosswordcheats.com>

logreduce

The LogReduce algorithm uses fuzzy logic to cluster messages together based on string and pattern similarity. Use the [LogReduce](#) button and operator to quickly assess activity patterns for things like a range of devices or traffic on a website.

About logreduce: Not supported in Dashboards.

save

Using the Save operator allows you to save the results of a query into the Sumo Logic file system. Later, you can use the lookup operator to access the saved data. The Save operator saves data in a simple format to a location you choose.

About save: Not supported in Dashboards.

Visualization

transpose Turn a list into a table in the Aggregates tab.

`transpose row [row fields] column [column fields]`

```
_sourc eCa teg ory =La bs/ Apa che /Access | timeslice 5m | count by _times lice, status
atus _code
```



By **TME520** (TME520)
cheatography.com/tme520/tme520.com

Published 4th September, 2017.
 Last updated 26th April, 2020.
 Page 8 of 14.

Sponsored by **CrosswordCheats.com**
 Learn to solve cryptic crosswords!
<http://crosswordcheats.com>