

splunk> .conf2017

Beyond “Regular” Regular Expressions

Cary Petterborg | Splunk Architect | LDS Church

August 8, 2017



Boilerplate

Forward-Looking Statements

During the course of this presentation, we may make forward-looking statements regarding future events or the expected performance of the company. We caution you that such statements reflect our current expectations and estimates based on factors currently known to us and that actual events or results could differ materially. For important factors that may cause actual results to differ from those contained in our forward-looking statements, please review our filings with the SEC.

The forward-looking statements made in this presentation are being made as of the time and date of its live presentation. If reviewed after its live presentation, this presentation may not contain current or accurate information. We do not assume any obligation to update any forward looking statements we may make. In addition, any information about our roadmap outlines our general product direction and is subject to change at any time without notice. It is for informational purposes only and shall not be incorporated into any contract or other commitment. Splunk undertakes no obligation either to develop the features or functionality described or to include any such feature or functionality in a future release.

Splunk, Splunk>, Listen to Your Data, The Engine for Machine Data, Splunk Cloud, Splunk Light and SPL are trademarks and registered trademarks of Splunk Inc. in the United States and other countries. All other brand names, product names, or trademarks belong to their respective owners. © 2017 Splunk Inc. All rights reserved.

My Disclaimer

During the course of this presentation, I may make references to my employer, The Church of Jesus Christ of Latter-day Saints. This should not be taken as an endorsement of Splunk or Splunk products by the LDS Church.



About Me

Who is Cary Petterborg

- ▶ Splunk user and administrator for 5.5 years
- ▶ Monitoring Engineer for 10 years
- ▶ Web developer for 23 years
- ▶ Software engineer for 37 years
- ▶ Many languages from assembly to Ruby
- ▶ Application development including Flight Sim, DB systems, and Web
- ▶ Works for the LDS Church in Salt Lake City
- ▶ Speaker at .conf 201[567]
- ▶ SplunkTrust Member 2018



Purpose

My purpose here today is to...

- ▶ Help you **control your data** instead of letting your data control you
- ▶ Regular expressions give you that control

Where's Waldo?

Can you easily pick out all the female Waldo's?



Picture courtesy of Albanpix.com

Regular Expressions help you...

- ▶ Find the distinctions within similar data
- ▶ Isolate the value properties from the noise
- ▶ Find the needles in the haystacks
- ▶ Break data into usable, constituent parts



Why Do I Like Regular Expressions?

- ▶ Using Regular Expressions since the mid 80's
- ▶ Started using regex with lex/yacc/sed/grep for software development
- ▶ Realized the power of regex quickly
- ▶ Taught classes on regex
- ▶ Love working with regex stuff in Splunk and other utilities
- ▶ **Regex is an important skill, and I want to share my knowledge**
- ▶ **Have Rex – Will Conquer**

One day, you too...

REGULAR EXPRESSION

1 MATCH - 9799 STEPS

```
/ ^\w{3}\s\d\d \d\d:\d\d:\d\d (?P<host>\w+).*Description=\s*(?P<descr>[\s\S]+20\d\d):[\s\S]*(MediaResourceList  
Name\s+:\s+(?P<media_res_list_name>\w+)[\s\S]*MediaResourceType\s+:\s+(?P<media_res_type>\d+)|RouteListName\  
s+:\s+(?P<route_list_name>[^\s;]+)[\s\S]*Reason=(?P<reason>\d+)[\s\S]*RouteGroups\((?P<route_grps>[^\s\)]+)\)|T  
imeStamp\s+:\s+(?P<timestamp>\d+\/\d+\/\d+\s+\d+:\d+\s+\w+)\s+LoginFrom\s+:\s+(?P<login_from>[^\s\)]+)\s+Inter  
face\s+:\s+(?P<interface>\w+)\s+UserID\s+:\s+(?P<user_id>\w+)\s+AppID\s+:\s+(?P<app_id>[\w\s]+)[\s\S]*C  
lusterID\s+:\s+(?P<cluster_id>[\w-]*)[\s\S]+NodeID\s+:\s+(?P<node_id>\w+)\s+Timestamp\s+:\s+(?P<ts>\w{3}  
[\s\S]*\w{3}\s+\d+\s+\d+:\d+:\d+\s+\w+\s+\d+)\s+::Status=(?P<status>\d+,[a-z]+)\s+Severity=(?P<severity>\w+)\s+  
Acknowledged=(?P<acked>\w+)\s+CUSTOMER=(?P<customer>[^\s\^]+)\s+Private IP Address=(?P<priv_ip>[^\s\^]+)\s+Default  
Alarm Name=(?P<def_alarm_name>[^\s\^]+)\s+Managed Object=(?P<object>[^\s\^]+)\s+Managed Object Type=(?P<obj_type>  
[^\s\^]+)\s+MODE=(?P<mode>[^\s;]+);Alarm ID=(?P<alarm_id>\d+)\s+Component=(?P<component>[^\s\^]+)\x00000
```

gmixXsuUAJ ?

TEST STRING

```
Jan 14 11:18:46 h32467 CPCMh32467: %local7-2-ALARM: 16$Description= Number of MediaResourceListExhausted events exceeds confi  
gured threshold during configured interval 5 within 60 minutes on cluster US-UT-DIR-Production. There are 6 MediaResourceLis  
tExhausted events (up to 30) received during the monitoring interval From Thu Jan 14 10:50:52 MST 2016 to Thu Jan 14 11:50:52  
MST 2016: MediaResourceListName : US-UT-DIR-Campus-NoVideo MediaResourceType : 2 AppID : Cisco CallManager ClusterID : US-UT  
-DIR-Production NodeID : dirvp211a Timestamp : Thu Jan 14 11:17:53 MST 201::Status=1,active^Severity=minor^Acknowledged=no^CU  
STOMER=Cisco Prime Collaboration^Private IP Address=10.308.5.22^Default Alarm Name=MediaListExhausted^Managed Object=10.308.5.  
22^Managed Object Type=Communications Manager^MODE=2;Alarm ID=659201239^Component=dirvp212a.wh.hamsterfarm.net\x00000
```

```
130 60.4 - - [07/Jan 18:10:57:153] "GET /c/
128.241.220.82 - - [07/Jan 18:10:57:1231] "
" 317 27.160.0.0 - - [07/Jan 18:10:56:156]
ows NT 5.1; SV1; - .NET CLR 1.1.4322" 468
itemId=EST-168product_id=RP-LI-02" 0
/buttercup-shopping.com/
opping=purchase&id=
/buttercup/case&id=
```

[Regular Expression Reference](#) [View in Search](#)

Preview Save

✓ 10 events (before 7/11/16 4:29:50.000 PM) Original search included: ? ☒ 20 per page ▾

atches

```
130 60.4 - - [07/Jan 18:10:57:153] "GET /c/
128.241.220.82 - - [07/Jan 18:10:57:1231] "
" 317 27.160.0.0 - - [07/Jan 18:10:56:156]
ows NT 5.1; SV1; - .NET CLR 1.1.4322" 468
itemId=EST-168product_id=RP-LI-02" 0
/buttercup-shopping.com/
opping=purchase&id=
/buttercup/case&id=
```



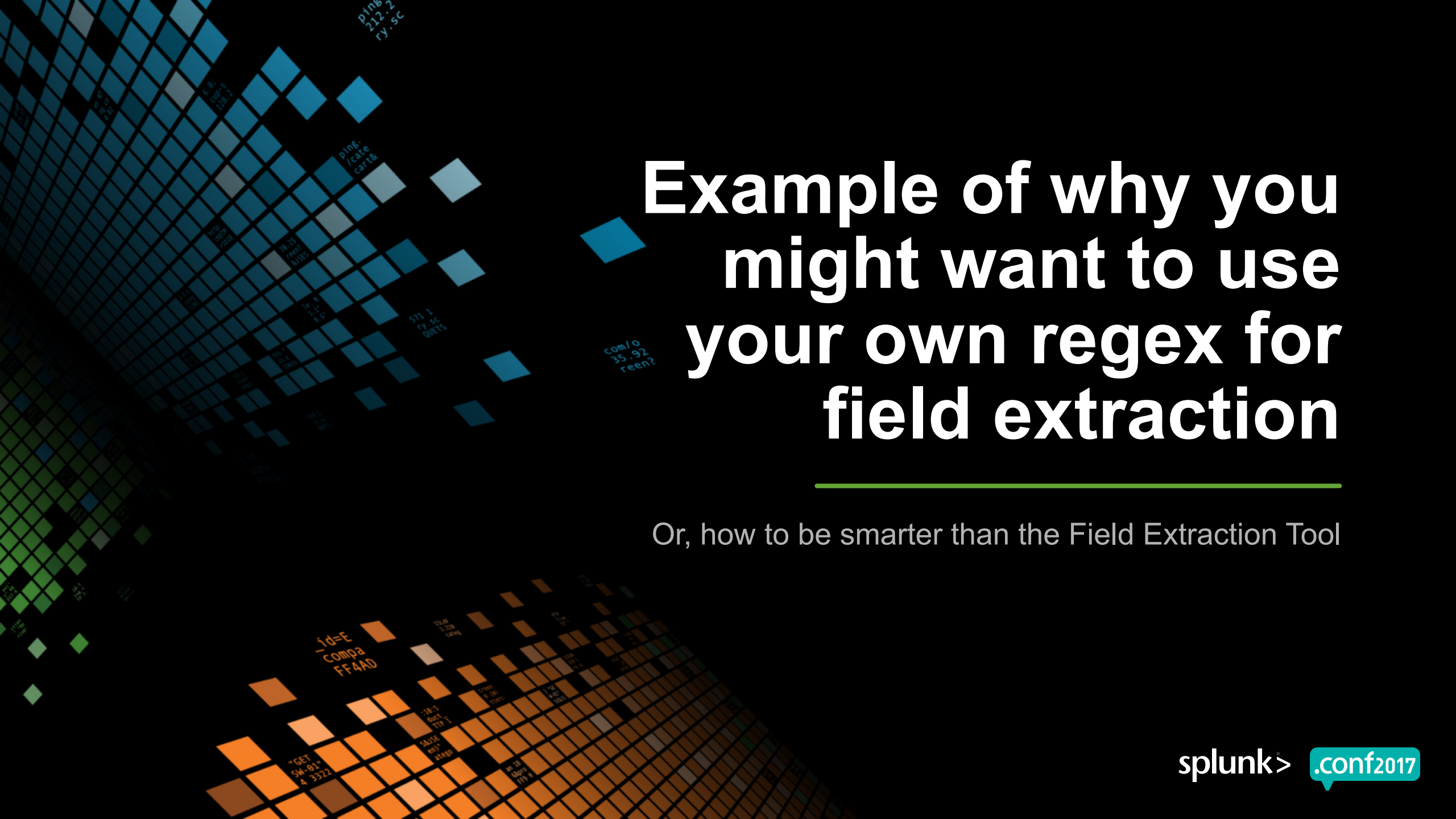

Splunk and Regular Expressions



Splunk Field Extraction Tool

Field Extraction Tool

- ▶ GUI tool in the web UI of Splunk
- ▶ Simple to use, and you can visually see the results of the regex on the events
- ▶ Pretty good *for a start*, but not *always* good for a final result
- ▶ Not able to optimize or do anything complex
- ▶ Makes mistakes if you don't have regular data



Example of why you might want to use your own regex for field extraction

Or, how to be smarter than the Field Extraction Tool

Multi-format Security Data in...

```

1 Mon Jun 06 2016 02:29:19 www1 sshd[3849]: Failed password for root from 128.241.220.82 port 2253 ssh2~
2 Mon Jun 06 2016 02:29:24 www1 sshd[4267]: Failed password for invalid user administrator from 128.241.220.82 port 1715 ssh2~
3 Mon Jun 06 2016 02:29:44 www1 sshd[5001]: Failed password for invalid user george from 128.241.220.82 port 2212 ssh2~
4 Mon Jun 06 2016 02:30:13 www1 sshd[1638]: pam_unix(sshd:session): session opened for user djohnson by (uid=0)~
5 Mon Jun 06 2016 02:30:13 www1 sshd[1480]: Failed password for invalid user yp from 128.241.220.82 port 2808 ssh2~
6 Mon Jun 06 2016 02:30:22 www1 sshd[2291]: Failed password for invalid user email from 128.241.220.82 port 4995 ssh2~
7 Mon Jun 06 2016 02:30:26 www1 sshd[4761]: Failed password for invalid user local from 128.241.220.82 port 1271 ssh2~
8 Mon Jun 06 2016 02:30:50 www1 sshd[4986]: Failed password for invalid user mysql from 128.241.220.82 port 2076 ssh2~
9 Mon Jun 06 2016 02:31:19 www1 sshd[81145]: pam_unix(sshd:session): session opened for user djohnson by (uid=0)~
10 Mon Jun 06 2016 02:31:19 www1 sshd[2021]: Failed password for myuan from 10.1.10.172 port 4468 ssh2~
11 Mon Jun 06 2016 02:31:28 www1 sshd[1155]: Failed password for invalid user yp from 10.1.10.172 port 1822 ssh2~
12 Mon Jun 06 2016 02:31:32 www1 sshd[1632]: Failed password for invalid user elena_andubasquet from 10.1.10.172 port 2074 ssh2~
13 Mon Jun 06 2016 02:31:54 www1 sshd[4333]: Failed password for root from 10.1.10.172 port 2772 ssh2~
14 Mon Jun 06 2016 02:32:00 www1 sudo: djohnson ; TTY=pts/0 ; PWD=/home/djohnson ; USER=root ; COMMAND=/bin/su~
15 Mon Jun 06 2016 02:32:00 www1 sshd[3697]: Failed password for invalid user whois from 10.1.10.172 port 1246 ssh2~
16 Mon Jun 06 2016 02:32:19 www1 sshd[5985]: Failed password for invalid user testuser from 10.1.10.172 port 2597 ssh2~

```


Simple FET extraction of Port

This is what the FET gives you:

Select Fields

Highlight one or more values in the sample event to create fields. You can indicate one value is required, meaning it must exist in an event for the regular expression

```
Wed Jun 08 2016 09:39:12 www1 sshd[4092]: Failed password for invalid user local from 109.169.32.135 port 2172 ssh2
```

Hide Regular Expression ▾

```
^(?:[^\.\n]*\.){3}\d+\s+\w+\s+(?P<port>\d+)
```

Regular Expression

Notes on Named Capture Groups

- ▶ `(?P<name>...)` === `(?<name>...)`
- ▶ The **P** is optional (came from **P**ython), but it is usually considered more correct
- ▶ Splunk FET will use `(?P<name>...)`, so why not make things similar?

BUT

- Do it the way you feel most comfortable

Goal: *user* from all entries using one regex

```
1 sshd[3697]: Failed password for invalid user whois from 10.1.10.172 port 124
1 sudo: djohnson ; TTY=pts/0 ; PWD=/home/djohnson ; USER=root ; COMMAND=/bin/
1 sshd[4333]: Failed password for root from 10.1.10.172 port 2772 ssh2
1 sshd[1632]: Failed password for invalid user elena_andubasquet from 10.1.10.
1 sshd[1155]: Failed password for invalid user yp from 10.1.10.172 port 1822 s
1 sshd[2021]: Failed password for myuan from 10.1.10.172 port 4468 ssh2
1 sshd[81145]: pam_unix(sshd:session): session opened for user djohnson by (ui
1 sshd[4986]: Failed password for invalid user mysql from 128.241.220.82 port
1 sshd[4761]: Failed password for invalid user local from 128.241.220.82 port
```


There is no user automatically extracted

i	Time	Event
▼	6/8/16 9:39:12.000 AM	Wed Jun 08 2016 09:39:12 www1 sshd[4092]: Failed password for invalid user local f
Event Actions ▼		
Type	☒	Field Value Actions
Selected	☒	host ▼ Carys-MacBook-Pro-2.local ▼
	☒	source ▼ secure.log ▼
	☒	sourcetype ▼ linux_secure ▼
Event	☐	pid ▼ 4092 ▼
	☐	process ▼ sshd ▼
Time +		_time ▼ 2016-06-08T09:39:12.000-06:00
Default	☐	index ▼ main ▼
	☐	linecount ▼ 1 ▼
	☐	punct ▼ ____:__:____:____:____ ▼
	☐	splunk_server ▼ Carys-MacBook-Pro-2.local ▼

FET Failed

Gets wrong values from some events

```
: Failed password for invalid user local from 109.169.32.135 port 2
]: Failed password for nsharp from 10.2.10.163 port 8317 ssh2
: Failed password for invalid user dean from 109.169.32.135 port 38
: Failed password for invalid user operator from 109.169.32.135 port
: Failed password for invalid user itmadmin from 109.169.32.135 port
: Failed password for mail from 84.34.159.23 port 1190 ssh2
: Failed password for sync from 84.34.159.23 port 2530 ssh2
: Failed password for invalid user inet from 10.3.10.46 port 1516 s
]: pam_unix(sshd:session): session closed for user myuan by (uid=0)
: Failed password for invalid user administrator from 10.3.10.46 port
```

```
130.60.4 - - [07/Jun 18:10:57:153] "GET /category.screen?category_id=GIFTS&product_id=F1-SW-03" 404 720 "http://buttercup-shopping.com/cart.do?action=view&itemId=EST-6&product_id=F1-SW-03"
128.241.220.82 - - [07/Jun 18:10:57:123] "GET /product.screen?product_id=FL-DSH-01&SESSIONID=5D55L7FF6ADFF9 HTTP 1.1" 404 3322 "http://buttercup-shopping.com/cart.do?action=purchase&itemId=EST-26&product_id=K0-CW-01"
317 27.160.0.0 - - [07/Jun 18:10:56:156] "GET /oldlink?item_id=EST-26&SESSIONID=5D55L9FF1ADFF3 HTTP 1.1" 200 1318 "http://buttercup-shopping.com/cart.do?action=changequantity&itemId=EST-18&product_id=AV-CB-01&SESSIONID=5D185L8FF3ADFF9"
: //buttercup-shopping.com/cart.do?action=purchase&itemId=EST-16&product_id=RP-LI-02" 468 125.17 14.11 "GET /category.screen?category_id=FLOWERS&SESSIONID=5D55L7FF6ADFF9 HTTP 1.1" 200 3885 "http://buttercup-shopping.com/cart.do?action=purchase&itemId=EST-26&product_id=K0-CW-01"
: //buttercup-shopping.com/cart.do?action=purchase&itemId=EST-26&product_id=K0-CW-01" 468 125.17 14.11 "GET /category.screen?category_id=FLOWERS&SESSIONID=5D55L7FF6ADFF9 HTTP 1.1" 200 3885 "http://buttercup-shopping.com/cart.do?action=purchase&itemId=EST-26&product_id=K0-CW-01"
```

Trying to add an additional line and extracting *user* doesn't work

Select Fields

```
Wed Jun 08 2016 09:39:12 www1 sshd[4092]: Failed password for invalid user local from 109.169.32.135 port 2172 ssh2
```

Wed Jun 08 2016 09:39:12 www1 sshd[38618]: Failed password for nsharp from 10.2.10.163 port 8317 ssh2

Preview

If you see incorrect results below, click an additional event to add it to the set of sample events. Highlight its values to improve the extraction. You can remove incorrect values in the next step.

Not pretty, not easy to change/fix, not efficient. So, let's fix it.

local

My tool of choice: regex101.com

regular expressions 101

@regex101 donate contact bug reports & feedback wiki

</>

SAVE & SHARE

save regex %S

FLAVOR

</> pcre (php) ✓

</> javascript

</> python

</> golang

TOOLS

code generator

regex debugger

REGULAR EXPRESSION

no match

/ insert your regular expression here /g

TEST STRING

insert your test string here

SWITCH TO UNIT TESTS

SUBSTITUTION

EXPLANATION

An explanation of your regex will be automatically generated as you type.

MATCH INFORMATION

Detailed match information will be displayed here automatically.

QUICK REFERENCE

splunk> .conf2017

Add the data and a regex

REGULAR EXPRESSION

:/ for (?P<user>\S+) from

TEST STRING

```
Mon Jun 06 2016 02:29:19 www1 sshd[3849]: Failed password for root from 128.2
Mon Jun 06 2016 02:29:24 www1 sshd[4267]: Failed password for invalid user ad
Mon Jun 06 2016 02:29:44 www1 sshd[5001]: Failed password for invalid user ge
Mon Jun 06 2016 02:30:13 www1 sshd[1638]: pam_unix(sshd:session): session ope
Mon Jun 06 2016 02:30:13 www1 sshd[1480]: Failed password for invalid user yp
Mon Jun 06 2016 02:30:22 www1 sshd[2291]: Failed password for invalid user er
Mon Jun 06 2016 02:30:26 www1 sshd[4761]: Failed password for invalid user lo
Mon Jun 06 2016 02:30:50 www1 sshd[4986]: Failed password for invalid user my
Mon Jun 06 2016 02:31:19 www1 sshd[81145]: pam_unix(sshd:session): session op
Mon Jun 06 2016 02:31:19 www1 sshd[2021]: Failed password for myuan from 10.1
Mon Jun 06 2016 02:31:28 www1 sshd[1155]: Failed password for invalid user yp
Mon Jun 06 2016 02:29:19 www1 sshd[3849]: Failed password for root from 128.2
Mon Jun 06 2016 02:29:24 www1 sshd[4267]: Failed password for invalid user ad
```

Refine the regex – better matches, but not all

REGULAR EXPRESSION

```
:/ for (invalid user )?(?P<user>\S+) from
```

TEST STRING

```
Mon Jun 06 2016 02:29:19 www1 sshd[3849]: Failed password for root from 128.241.220.82 port 2253
Mon Jun 06 2016 02:29:24 www1 sshd[4267]: Failed password for invalid user administrator from 12
Mon Jun 06 2016 02:29:44 www1 sshd[5001]: Failed password for invalid user george from 128.241.2
Mon Jun 06 2016 02:30:13 www1 sshd[1638]: pam_unix(sshd:session): session opened for user djohns
Mon Jun 06 2016 02:30:13 www1 sshd[1480]: Failed password for invalid user yp from 128.241.220.8
Mon Jun 06 2016 02:30:22 www1 sshd[2291]: Failed password for invalid user email from 128.241.22
Mon Jun 06 2016 02:30:26 www1 sshd[4761]: Failed password for invalid user local from 128.241.22
Mon Jun 06 2016 02:30:50 www1 sshd[4986]: Failed password for invalid user mysql from 128.241.22
Mon Jun 06 2016 02:31:19 www1 sshd[81145]: pam_unix(sshd:session): session opened for user djohr
Mon Jun 06 2016 02:31:19 www1 sshd[2021]: Failed password for myuan from 10.1.10.172 port 4468 s
Mon Jun 06 2016 02:31:28 www1 sshd[1155]: Failed password for invalid user yp from 10.1.10.172 p
Mon Jun 06 2016 02:29:19 www1 sshd[3849]: Failed password for root from 128.241.220.82 port 2253
Mon Jun 06 2016 02:29:24 www1 sshd[4267]: Failed password for invalid user administrator from 12
Mon Jun 06 2016 02:29:44 www1 sshd[5001]: Failed password for invalid user george from 128.241.2
Mon Jun 06 2016 02:30:13 www1 sshd[1638]: pam_unix(sshd:session): session opened for user djohns
Mon Jun 06 2016 02:30:13 www1 sshd[1480]: Failed password for invalid user yp from 128.241.220.8
Mon Jun 06 2016 02:30:22 www1 sshd[2291]: Failed password for invalid user email from 128.241.22
Mon Jun 06 2016 02:30:26 www1 sshd[4761]: Failed password for invalid user local from 128.241.22
Mon Jun 06 2016 02:30:50 www1 sshd[4986]: Failed password for invalid user mysql from 128.241.22
Mon Jun 06 2016 02:31:19 www1 sshd[81145]: pam_unix(sshd:session): session opened for user diohr
```

Refine the regex again – almost there

REGULAR EXPRESSION

```
:/ for ((invalid user)|(user ))?(?P<user>\S+) (from|by)
```

TEST STRING

```
Mon Jun 06 2016 02:29:19 www1 sshd[3849]: Failed password for root from 128.241.220.82 port 2253 ssh2
Mon Jun 06 2016 02:29:24 www1 sshd[4267]: Failed password for invalid user administrator from 128.241.220.82 port 2253 ssh2
Mon Jun 06 2016 02:29:44 www1 sshd[5001]: Failed password for invalid user george from 128.241.220.82 port 2253 ssh2
Mon Jun 06 2016 02:30:13 www1 sshd[1638]: pam_unix(sshd:session): session opened for user djohnson by (uid=0)
Mon Jun 06 2016 02:30:13 www1 sshd[1480]: Failed password for invalid user yp from 128.241.220.82 port 2253 ssh2
Mon Jun 06 2016 02:30:22 www1 sshd[2291]: Failed password for invalid user email from 128.241.220.82 port 2253 ssh2
Mon Jun 06 2016 02:30:24 www1 sudo: djohnson ; TTY=pts/0 ; PWD=/home/djohnson ; USER=root ; COMMAND=/bin/bash
Mon Jun 06 2016 02:30:26 www1 sshd[4761]: Failed password for invalid user local from 128.241.220.82 port 2253 ssh2
Mon Jun 06 2016 02:30:50 www1 sshd[4986]: Failed password for invalid user mysql from 128.241.220.82 port 2253 ssh2
Mon Jun 06 2016 02:31:19 www1 sshd[81145]: pam_unix(sshd:session): session opened for user djohnson by (uid=0)
Mon Jun 06 2016 02:31:19 www1 sshd[2021]: Failed password for myuan from 10.1.10.172 port 4468 ssh2
Mon Jun 06 2016 02:31:28 www1 sshd[1155]: Failed password for invalid user yp from 10.1.10.172 port 1822 ssh2
Mon Jun 06 2016 02:29:19 www1 sshd[3849]: Failed password for root from 128.241.220.82 port 2253 ssh2
Mon Jun 06 2016 02:29:24 www1 sshd[4267]: Failed password for invalid user administrator from 128.241.220.82 port 2253 ssh2
Mon Jun 06 2016 02:29:44 www1 sshd[5001]: Failed password for invalid user george from 128.241.220.82 port 2253 ssh2
```

130.60.4 - - [07/Jun 2016 18:10:57:153] "GET /category.screen?category_id=GIFTS&SESSIONID=SD5SL9FF1ADFF3 HTTP 1.1" 404 720 "http://buttercup-shopping.com/cart.do?action=view&itemId=EST-6&product_id=F1-SW-03" "Opera/9.80 (Win32; x64; rv:26.0) Gecko/20100101 Firefox/26.0"
128.241.220.82 - - [07/Jun 2016 18:10:57:123] "GET /product.screen?product_id=FL-DSH-01&SESSIONID=SD5SL9FF1ADFF3 HTTP 1.1" 200 1318 "http://buttercup-shopping.com/cart.do?action=changequantity&itemId=EST-18&product_id=AV-CB-01&SESSIONID=SD18SL9FF1ADFF3 HTTP 1.1" 200 385 "http://buttercup-shopping.com/cart.do?action=remove&itemId=EST-6&product_id=F1-SW-03" "Opera/9.80 (Win32; x64; rv:26.0) Gecko/20100101 Firefox/26.0"
130.60.4 - - [07/Jun 2016 18:10:57:153] "GET /category.screen?category_id=GIFTS&SESSIONID=SD5SL9FF1ADFF3 HTTP 1.1" 404 720 "http://buttercup-shopping.com/cart.do?action=view&itemId=EST-6&product_id=F1-SW-03" "Opera/9.80 (Win32; x64; rv:26.0) Gecko/20100101 Firefox/26.0"
128.241.220.82 - - [07/Jun 2016 18:10:57:123] "GET /product.screen?product_id=FL-DSH-01&SESSIONID=SD5SL9FF1ADFF3 HTTP 1.1" 200 1318 "http://buttercup-shopping.com/cart.do?action=changequantity&itemId=EST-18&product_id=AV-CB-01&SESSIONID=SD18SL9FF1ADFF3 HTTP 1.1" 200 385 "http://buttercup-shopping.com/cart.do?action=remove&itemId=EST-6&product_id=F1-SW-03" "Opera/9.80 (Win32; x64; rv:26.0) Gecko/20100101 Firefox/26.0"

Four different formats – all four user field types found!

REGULAR EXPRESSION

```
/((for ((invalid user)|(user ))?)|(sudo: ))(?P<user>\S+) (from|by)?
```

TEST STRING

```
Mon Jun 06 2016 02:29:19 www1 sshd[3849]: Failed password for root from 128.241.220.82 port 2253 ssh2
Mon Jun 06 2016 02:29:24 www1 sshd[4267]: Failed password for invalid user administrator from 128.241.220.82 port 2253 ssh2
Mon Jun 06 2016 02:29:44 www1 sshd[5001]: Failed password for invalid user george from 128.241.220.82 port 2253 ssh2
Mon Jun 06 2016 02:30:13 www1 sshd[1638]: pam_unix(sshd:session): session opened for user djohnson by (uid=0)
Mon Jun 06 2016 02:30:13 www1 sshd[1480]: Failed password for invalid user yp from 128.241.220.82 port 2253 ssh2
Mon Jun 06 2016 02:30:22 www1 sshd[2291]: Failed password for invalid user email from 128.241.220.82 port 2253 ssh2
Mon Jun 06 2016 02:30:24 www1 sudo: djohnson ; TTY=pts/0 ; PWD=/home/djohnson ; USER=root ; COMMAND=/bin/bash
Mon Jun 06 2016 02:30:26 www1 sshd[4761]: Failed password for invalid user local from 128.241.220.82 port 2253 ssh2
Mon Jun 06 2016 02:30:50 www1 sshd[4986]: Failed password for invalid user mysql from 128.241.220.82 port 2253 ssh2
Mon Jun 06 2016 02:31:19 www1 sshd[81145]: pam_unix(sshd:session): session opened for user djohnson by (uid=0)
Mon Jun 06 2016 02:31:19 www1 sshd[2021]: Failed password for myuan from 10.1.10.172 port 4468 ssh2
Mon Jun 06 2016 02:31:28 www1 sshd[1155]: Failed password for invalid user yp from 10.1.10.172 port 1822 ssh2
Mon Jun 06 2016 02:29:19 www1 sshd[3849]: Failed password for root from 128.241.220.82 port 2253 ssh2
```

Back to the FET – use our regex

Paste this regex into the field that the ugly, single-format regex was in:

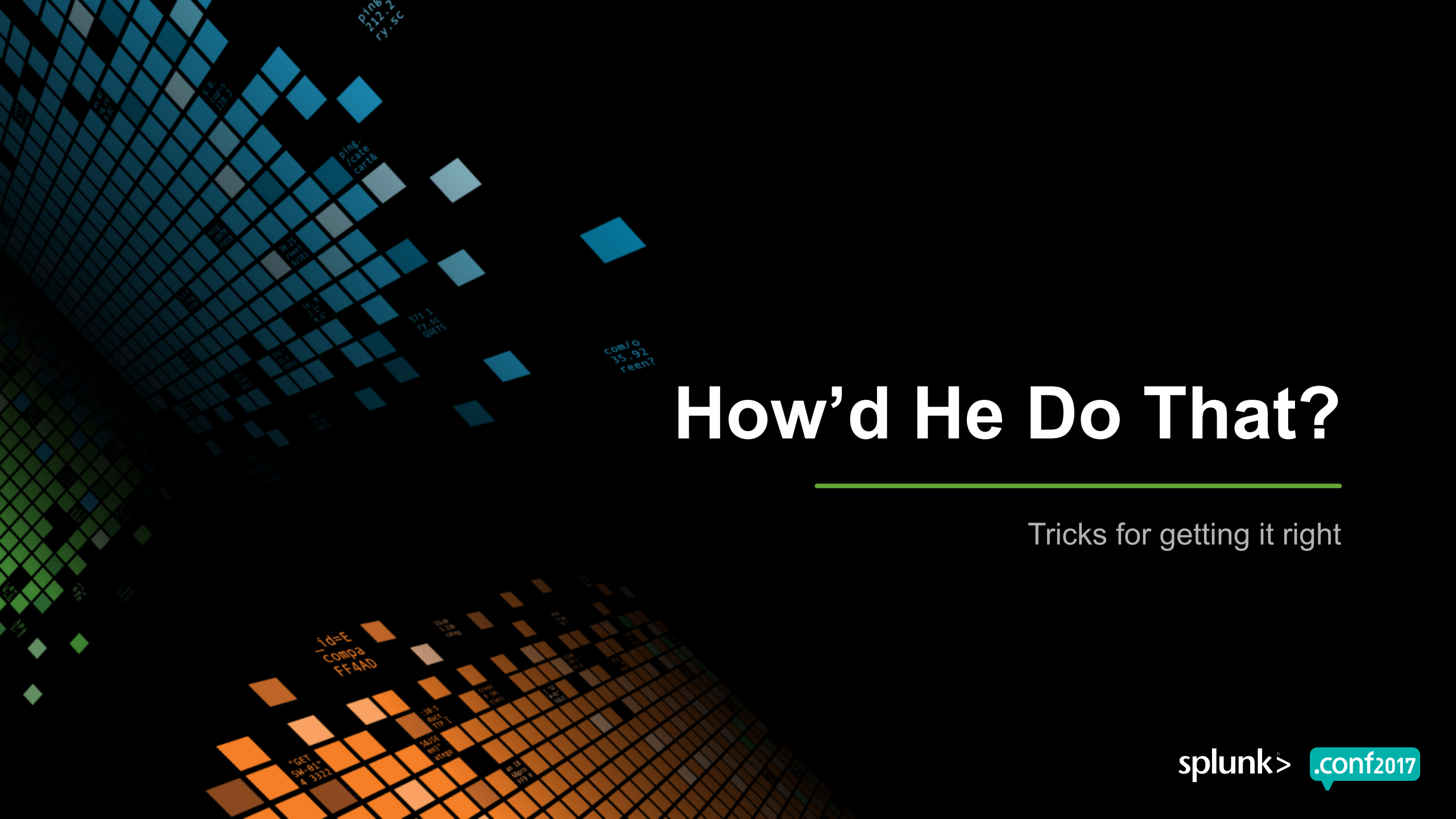
Regular Expression

```
((for ((invalid user ))|(user ))?)(sudo: )?(P<user>\S+) (from|by)?
```

...and the results are MUCH better

```
www1 sshd[3965]: Failed password for invalid user testuser from 10.1.10.172 port 1
www1 sshd[3697]: Failed password for invalid user whois from 10.1.10.172 port 1
www1 sudo: djohnson ; TTY=pts/0 ; PWD=/home/djohnson ; USER=root ; COMMAND=/bi
www1 sshd[4333]: Failed password for root from 10.1.10.172 port 2772 ssh2
www1 sshd[1632]: Failed password for invalid user elena_andubasquet from 10.1.1
www1 sshd[1155]: Failed password for invalid user yp from 10.1.10.172 port 1822
www1 sshd[2021]: Failed password for myuan from 10.1.10.172 port 4468 ssh2
www1 sshd[81145]: pam_unix(sshd:session): session opened for user djohnson by (
www1 sshd[4986]: Failed password for invalid user mysql from 128.241.220.82 por
www1 sshd[4761]: Failed password for invalid user local from 128.241.220.82 por
www1 sshd[2291]: Failed password for invalid user email from 128.241.220.82 por
www1 sshd[1480]: Failed password for invalid user yp from 128.241.220.82 port 2
www1 sshd[1638]: pam_unix(sshd:session): session opened for user djohnson by (u
www1 sshd[50011]: Failed password for invalid user george from 128.241.220.82 po
```

```
130.60.4 - - [07/Jan 18:10:57:153] "GET /category.screen?category_id=GIFTS&SESSIONID=5D5SLAFF10ADFF10 HTTP 1.1" 404 720 "http://buttercup-shopping.com/category.screen?category_id=GIFTS"
128.241.220.82 - - [07/Jan 18:10:57:123] "GET /product.screen?product_id=FL-DSH-01&SESSIONID=5D5SL7FF6ADFF9 HTTP 1.1" 404 3322 "http://buttercup-shopping.com/cart.do?action=purchase&item_id=EST-26&product_id=FL-DSH-01"
317 27.160.0.0 - - [07/Jan 18:10:56:156] "GET /oldlink?item_id=EST-26&SESSIONID=5D5SL9FF1ADFF3 HTTP 1.1" 200 1318 "http://buttercup-shopping.com/cart.do?action=changequantity&item_id=EST-18&product_id=AV-CB-01&SESSIONID=5D185L8FF3ADFF9 HTTP 1.1"
www1 sshd[50011]: Failed password for invalid user george from 128.241.220.82 po
```



How'd He Do That?

Tricks for getting it right

One named capture group with a single name

More than one instance of the same name will fail

- ▶ (for invalid user (?P<user>\S+))|(for (?P<user>\S+))
- ▶ Capture group names must be unique:

EXPLANATION

▼ / (for invalid user (?P<user>\S+))|(for (?P<user>\S+)) / g

All the errors detected are listed below, from left to right, as they appear in the pattern.

(?P<user> A subpattern name must be unique

) A subpattern name must be unique

How do you eat an elephant?

Bite by bite is better than trying to stuff the whole elephant in your mouth at once

- ▶ Start with one format
- ▶ Try to find similarities and differences between the formats
- ▶ Add a new format to your data and check your updated regex
- ▶ Keep a copy of the last one that worked!!
- ▶ Add additional formats and check ALL matches for ALL examples

```
130.60.4 - - [07/Jan 18:10:57:123] "GET /category.screen?category_id=GIFTS&SESSIONID=5D5SLAFF10ADFF10 HTTP 1.1" 404 720 "http://buttercup-shopping.com/cart.do?action=view&itemId=EST-6&product_id=FI-SW-03"
128.241.220.82 - - [07/Jan 18:10:57:123] "GET /product.screen?product_id=FL-DSH-01&SESSIONID=5D5SL7FF6ADFF9 HTTP 1.1" 404 3322 "http://buttercup-shopping.com/cart.do?action=purchase&itemId=EST-26&product_id=KQ-CW-01"
317 27.160.0.0 - - [07/Jan 18:10:57:153] "GET /category.screen?category_id=FLOWERS&SESSIONID=5D5SL8FF1ADFF3 HTTP 1.1" 200 2423 "http://buttercup-shopping.com/cart.do?action=purchase&itemId=EST-26&product_id=KQ-CW-01"
ows NT 5.1; SV1; .NET CLR 1.1.4322" 468 125.17 14 - - [07/Jan 18:10:57:153] "GET /category.screen?category_id=FLOWERS&SESSIONID=5D5SL8FF1ADFF3 HTTP 1.1" 200 2423 "http://buttercup-shopping.com/cart.do?action=purchase&itemId=EST-26&product_id=KQ-CW-01"
itemId=EST-16&product_id=RP-LI-02" 468 125.17 14 - - [07/Jan 18:10:57:153] "GET /category.screen?category_id=FLOWERS&SESSIONID=5D5SL8FF1ADFF3 HTTP 1.1" 200 2423 "http://buttercup-shopping.com/cart.do?action=purchase&itemId=EST-26&product_id=KQ-CW-01"
toaction=purchase&itemId=EST-26&product_id=KQ-CW-01" 468 125.17 14 - - [07/Jan 18:10:57:153] "GET /category.screen?category_id=FLOWERS&SESSIONID=5D5SL8FF1ADFF3 HTTP 1.1" 200 2423 "http://buttercup-shopping.com/cart.do?action=purchase&itemId=EST-26&product_id=KQ-CW-01"
opping.com/cart.do?action=purchase&itemId=EST-26&product_id=KQ-CW-01" 468 125.17 14 - - [07/Jan 18:10:57:153] "GET /category.screen?category_id=FLOWERS&SESSIONID=5D5SL8FF1ADFF3 HTTP 1.1" 200 2423 "http://buttercup-shopping.com/cart.do?action=purchase&itemId=EST-26&product_id=KQ-CW-01"
http://buttercup-shopping.com/cart.do?action=purchase&itemId=EST-26&product_id=KQ-CW-01" 468 125.17 14 - - [07/Jan 18:10:57:153] "GET /category.screen?category_id=FLOWERS&SESSIONID=5D5SL8FF1ADFF3 HTTP 1.1" 200 2423 "http://buttercup-shopping.com/cart.do?action=purchase&itemId=EST-26&product_id=KQ-CW-01"
```

Alternate: Do one for each format

Then you can try combining

- ▶ This can be more difficult with simpler regexes
- ▶ Can be easier for more complex regexes
- ▶ Combine two of the regexes that are similar
- ▶ Try to keep the things that are the same in both, making the changes to the original only where there is a difference
- ▶ Remember – **ONE instance of a name per regex**

Make it easier to come back to later

Make it easier to come back to later

((for ((invalid user)|(user))?)(sudo:))(?P<user>\S+) (from|by)?

- ▶ Using parentheses for clarity is helpful:
- ▶ They make it possible to see the separate parts and their relationships with each other
- ▶ Don't overdo the parentheses

Use the Best Character Class

Use the right tool for the job

- ▶ Sometimes a field regex must be able to match data that hasn't been seen in the data yet, so in this case **be as general as possible**
- ▶ In the previous example, the **\S** is best because **\s** will be the delimiter (a space in this case) because you want to catch any potential case that you don't see in the data, *yet*.

$\backslash S_+$

- If you have a delimiter that you can count on, use something like this to match the field value (**in this case be specific** about what it is NOT):

$$[\wedge, +]$$

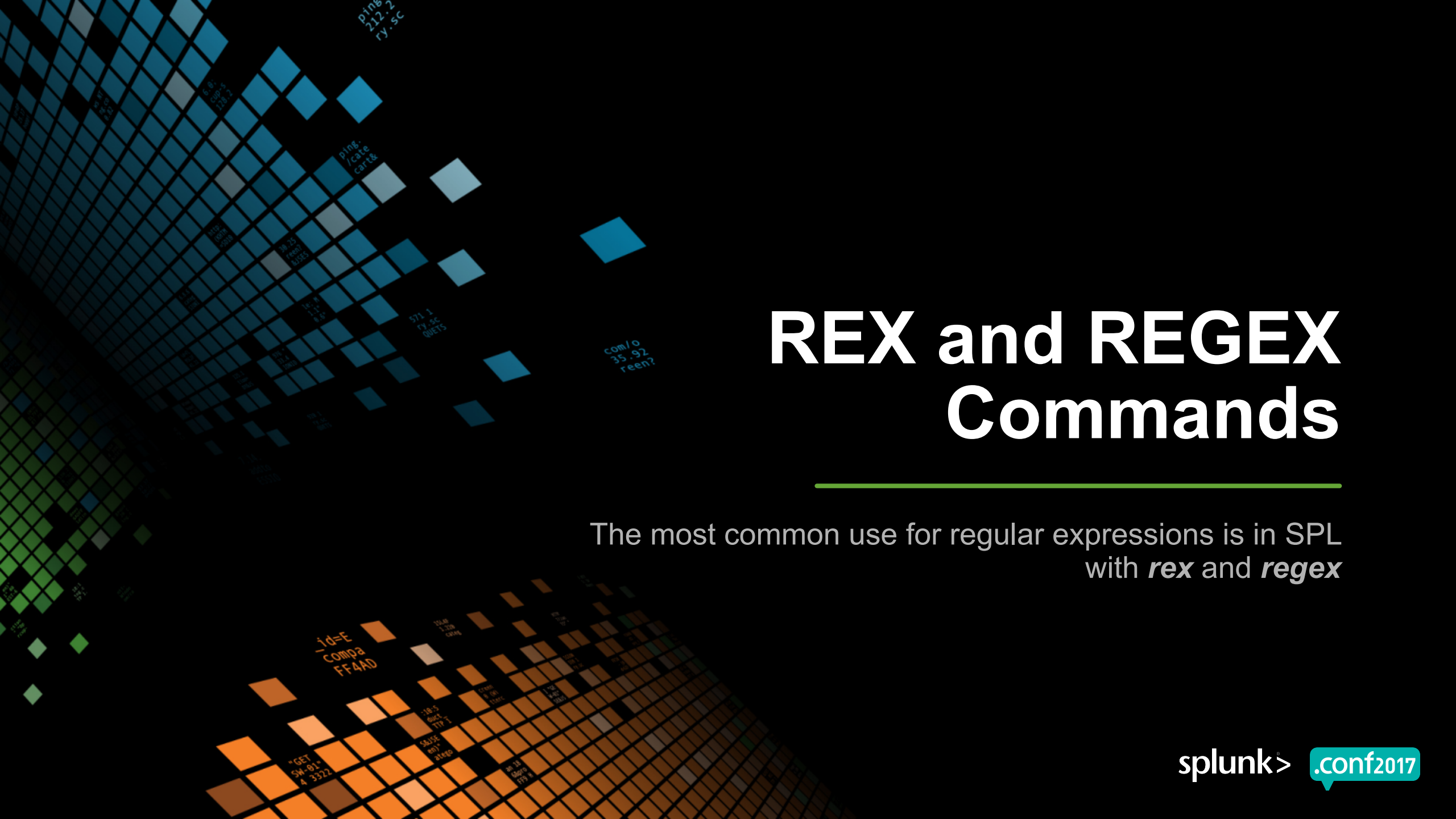
Just because your sample data doesn't have a particular character in it, that doesn't mean it never will. Examples:

- ▶ Usernames – alphanumerics + what?
 - Dash – Underscore - Other characters - Are you sure?
- ▶ Filenames – could you have a space?
 - C:\Program Files\My Application\

If you have a definite delimiter, take advantage

Examples:

- Data: "contents of quoted string"
 - Use: "(?P<contents>[^\"]+)"
- Data: User:carypetterborg Dept:ICS
 - Use: User:(?P<user>\S+)\s
- Data: Salt Lake City, Utah 84117-6403
 - Use: ^(?P<city>[^,]+),\s+(?P<state>.\s+)\s+(?P<zip>[-\d]*)\$



REX and REGEX Commands

The most common use for regular expressions is in SPL
with *rex* and *regex*

When to Use REX

When you:

- ▶ **don't always** want to extract the data
- ▶ want to extract data from a field that is already extracted
- ▶ don't have access to field extractions (permissions, etc.)
- ▶ require doing multiple, disparate regular expressions
- ▶ are in a hurry or you are doing a proof-of-concept

REX Example

```
index=voice sourctype=voice* Description | rex "Description=(?P<description>[^\^]+)"
| rex field=description "From (?P<start>.+ ) to (?P<end>.+?):\s"
```

Aug 2 20:32:28 I13772 CPCM13772: %local7-2-ALARM: 16\$Description= Number of AuthenticationFailed events exceeds configured threshold during configured interval of time 1 within 3 minutes on cluster StandAloneCluster. There are 2 AuthenticationFailed events (up to 30) received during the monitoring interval From **Wed Aug 03 10:25:00 PHT 2016** to **Wed Aug 03 10:28:00 PHT 2016**: TimeStamp : 8/3/16 10:26 AM LoginFrom : 172.12.34.40 Interface : VMREST UserID : JacobMD AppID : Cisco Tomcat ClusterID : NodeID : APPHMANAOVM001 TimeStamp : Wed Aug 03 10:26:13 PHT 2016 TimeStam::Status=2,cleared^Severity=minor^Acknowledged=no^CUSTOMER=Cisco Prime Collaboration^Private IP Address=172.16.17.24^Default Alarm Name=AuthenticationFailed^Managed Object=10.160.17.24^Managed Object Type=Unity Connection^MODE=2;Alarm ID=343815480^Component=10.160.17.24\x00000

REX Commands

It takes two:

- ▶ index=voice sourcetype=voice* Description
 - | rex "Description=(?P<description>[^\^]+)"
 - | rex field=description "From (?P<start>.+) to (?P<end>.+?):\s"
- ▶ SYNTAX:
 - | rex [field=*fieldname*] "regex"
- ▶ Also available:
 - | rex mode=sed

First rex – get the description

REGULAR EXPRESSION

1 match, 23 steps (~5ms)

`/ Description=(?P<description>[^\^]+)`

/g

TEST STRING

SWITCH TO UNIT TESTS ▶

```
Aug  2 20:32:28 l13772 CPCML13772: %local7-2-ALARM: 16$Description= Number of
AuthenticationFailed events exceeds configured threshold during configured interval of
time 1 within 3 minutes on cluster StandAloneCluster. There are 2
AuthenticationFailed events (up to 30) received during the monitoring interval From Wed
Aug 03 10:25:00 PHT 2016 to Wed Aug 03 10:28:00 PHT 2016:   TimeStamp : 8/3/16 10:26 AM
LoginFrom : 172.12.34.40 Interface : VMREST UserID : JacobMD AppID : Cisco Tomcat
ClusterID : NodeID : APPHMANAOVM001 TimeStamp : Wed Aug 03 10:26:13 PHT 2016
TimeStam::Status=2,cleared^Severity=minor^Acknowledged=no^CUSTOMER=Cisco Prime
Collaboration^Private IP Address=172.16.17.24^Default Alarm
Name=AuthenticationFailed^Managed Object=10.160.17.24^Managed Object Type=Unity
Connection^MODE=2;Alarm ID=343815480^Component=10.160.17.24\x00000
```

Second rex – get the Start and End

REGULAR EXPRESSION1 match, 1016 steps (~16ms)

`From (?P<start>.+ ) to (?P<end>.+?):\s`/g

TEST STRINGSWITCH TO UNIT TESTS ▶

Number of AuthenticationFailed events exceeds configured threshold during configured interval of time 1 within 3 minutes on cluster StandAloneCluster. There are 2 AuthenticationFailed events (up to 30) received during the monitoring interval From Wed Aug 03 10:25:00 PHT 2016 to Wed Aug 03 10:28:00 PHT 2016: TimeStamp : 8/3/16 10:26 AM LoginFrom : 172.12.34.40 Interface : VMREST UserID : JacobMD AppID : Cisco Tomcat ClusterID : NodeID : APPHMANAOVM001 TimeStamp : Wed Aug 03 10:26:13 PHT 2016 TimeStam::Status=2,cleared

When to use REGEX

- ▶ To filter out events/data that you don't want included in the pipeline
- ▶ This is like **search** on steroids, but doesn't replace **search**
- ▶ Only used as a filter

```
130.60.4 - - [07/Jun 18:10:57:153] "GET /category.screen?category_id=GIFTS&SESSIONID=5015L4FF10ADFF10 HTTP 1.1" 404 720 "http://buttercup-shopping.com/cart.do?action=view&itemId=EST-6&product_id=FI-SW-03" "Mozilla/4.0" "Opera/9.20"
128.241.220.82 - - [07/Jun 18:10:57:123] "GET /product.screen?product_id=FL-DSH-01&SESSIONID=5035L7FF6ADFF0 HTTP 1.1" 404 3322 "http://buttercup-shopping.com/category.screen?category_id=GIFTS" "Mozilla/4.0" "Opera/9.20"
317 27.160.0.0 - - [07/Jun 18:10:56:156] "GET /oldlink?item_id=EST-26&SESSIONID=5055L9FF1ADFF3 HTTP 1.1" 200 1318 "http://buttercup-shopping.com/cart.do?action=changequantity&itemId=EST-18&product_id=AV-CB-01&SESSIONID=5015L4FF10ADFF10" "Mozilla/4.0" "Opera/9.20"
item_id=EST-16&product_id=RP-LI-02" 468 125.17 14.189] "GET /category.screen?category_id=FLOWERS&SESSIONID=5035L7FF6ADFF0 HTTP 1.1" 200 3885 "http://buttercup-shopping.com/category.screen?category_id=FLOWERS" "Mozilla/4.0" "Opera/9.20"
toaction=purchase&is.com/ol-02" "0-189] "GET /category.screen?category_id=FLOWERS&SESSIONID=5035L7FF6ADFF0 HTTP 1.1" 200 3885 "http://buttercup-shopping.com/category.screen?category_id=FLOWERS" "Mozilla/4.0" "Opera/9.20"
opping.com/category.screen?category_id=FLOWERS&SESSIONID=5035L7FF6ADFF0 HTTP 1.1" 200 3885 "http://buttercup-shopping.com/category.screen?category_id=FLOWERS" "Mozilla/4.0" "Opera/9.20"
```

Regex example

Only get events with internal addresses

New Search

sourcetype="linux_secure" | regex "10\\.\\d+\\.\\d+\\.\\d+" |

289 events (before 7/17/17 1:20:53.000 PM) No Event Sampling

Events (289) Patterns Statistics Visualization

Format Timeline Zoom Out Zoom to Selection Deselect

1 hour per column

List Format 20 Per Page

< Prev 1 2 3 4 5 6 7 8 9 ... Next >

< Hide Fields All Fields

Selected Fields

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

Interesting Fields

- # date_hour 17
- # date_mday 3
- # date_minute 55
- a date_month 1
- # date_second 59
- a date_wday 3
- # date_year 1
- a date_zone 1
- a index 1

#	Time	Event
>	6/8/16 9:39:12.000 AM	Wed Jun 08 2016 09:39:12 www1 sshd[38618]: Failed password for nsharp from 10.2.10.163 port 8317 ssh2 host = Carys-MacBook-Pro-2.local source = secure.log sourcetype = linux_secure
>	6/8/16 2:14:45.000 AM	Wed Jun 08 2016 02:14:45 www3 sshd[3957]: Failed password for invalid user inet from 10.3.10.46 port 1516 ssh2 host = Carys-MacBook-Pro-2.local source = secure.log sourcetype = linux_secure
>	6/8/16 2:14:41.000 AM	Wed Jun 08 2016 02:14:41 www3 sshd[4461]: Failed password for invalid user administrator from 10.3.10.46 port 2108 ssh2 host = Carys-MacBook-Pro-2.local source = secure.log sourcetype = linux_secure
>	6/8/16 2:14:23.000 AM	Wed Jun 08 2016 02:14:23 www3 sshd[3256]: Failed password for invalid user jetty from 10.3.10.46 port 3759 ssh2 host = Carys-MacBook-Pro-2.local source = secure.log sourcetype = linux_secure
>	6/8/16 2:13:52.000 AM	Wed Jun 08 2016 02:13:52 www3 sshd[4725]: Failed password for invalid user ubuntu from 10.3.10.46 port 4940 ssh2 host = Carys-MacBook-Pro-2.local source = secure.log sourcetype = linux_secure
>	6/8/16 2:13:48.000 AM	Wed Jun 08 2016 02:13:48 www3 sshd[5405]: Failed password for invalid user oracle from 10.3.10.46 port 4544 ssh2 host = Carys-MacBook-Pro-2.local source = secure.log sourcetype = linux_secure
>	6/8/16 2:13:48.000 AM	Wed Jun 08 2016 02:13:48 www3 sshd[68976]: Accepted password for djohnson from 10.3.10.46 port 1790 ssh2 host = Carys-MacBook-Pro-2.local source = secure.log sourcetype = linux_secure

Breakdown

- ▶ Search:
`sourcetype=linux_secure | regex "10\.\d+\.\d+\.\d+"`
- ▶ Only internal (10.*) IP addresses make it through the **regex** filter
- ▶ Search produces events, regex then limits those results passed on through the pipeline by a fancy regular expression
- ▶ *Yes, there are other ways to do this, but this is a regex example*

Rex vs Regex

- ▶ Use **rex** to extract fields
- ▶ Use **regex** to limit results
- ▶ Yes, you can use them in the same search:

```
sourcetype=linux_secure | rex "from (?P<src_ip>\d+\.\d+\.\d+\.\d+)" | regex  
src_ip="(?!10)\.\d+\.\d+\.\d+"
```



Index Time Regular Expression Usage

The Problem

- ▶ You can't index Social Security Numbers
- ▶ How do you distinguish a Social Security Number from other numbers?
- ▶ Obfuscate ONLY SSNs, but leave other things alone.

130.60.4 - - [07/Jun 18:10:57:153] "GET /category.screen?category_id=GIFTS&SESSIONID=5D5SLAFF10ADFF10 HTTP 1.1" 404 720 "http://buttercup-shopping.com/cart.do?action=view&itemId=EST-6&product_id=F1-SW-03"
128.241.220.82 - - [07/Jun 18:10:57:123] "GET /product.screen?product_id=FL-DSH-01&SESSIONID=5D5SL7FF6ADFF9 HTTP 1.1" 404 3322 "http://buttercup-shopping.com/category.screen?category_id=GIFTS&SESSIONID=5D5SLAFF10ADFF10 HTTP 1.1"
317 27.160.0.0 - - [07/Jun 18:10:57:123] "GET /product.screen?product_id=FL-DSH-01&SESSIONID=5D5SL7FF6ADFF9 HTTP 1.1" 200 1318 "http://buttercup-shopping.com/cart.do?action=purchase&itemId=EST-26&product_id=F1-SW-03"
ows NT 5.1; SV1; .NET CLR 1.1.4322" 468 125.17 14.189 "GET /category.screen?category_id=FLOWERS&SESSIONID=5D5SL8FF1ADFF3 HTTP 1.1" 200 3885 "http://buttercup-shopping.com/category.screen?category_id=FLOWERS&SESSIONID=5D5SL8FF1ADFF3 HTTP 1.1"
itemId=EST-16&product_id=RP-LI-02" 468 125.17 14.189 "GET /category.screen?category_id=FLOWERS&SESSIONID=5D5SL8FF1ADFF3 HTTP 1.1" 200 3885 "http://buttercup-shopping.com/category.screen?category_id=FLOWERS&SESSIONID=5D5SL8FF1ADFF3 HTTP 1.1"
do?action=purchase&itemId=EST-16&product_id=RP-LI-02" 468 125.17 14.189 "GET /category.screen?category_id=FLOWERS&SESSIONID=5D5SL8FF1ADFF3 HTTP 1.1" 200 3885 "http://buttercup-shopping.com/category.screen?category_id=FLOWERS&SESSIONID=5D5SL8FF1ADFF3 HTTP 1.1"
buttercup-shopping.com/category.screen?category_id=FLOWERS&SESSIONID=5D5SL8FF1ADFF3 HTTP 1.1" 200 3885 "http://buttercup-shopping.com/category.screen?category_id=FLOWERS&SESSIONID=5D5SL8FF1ADFF3 HTTP 1.1"

SSN vs Phone

Regex distinctions

SSN

▶ 123-45-6789

► $\text{d}\{3\} - \text{d}\{2\} - \text{d}\{4\}$

Phone #

▶ 800-123-4567

► $\{d_3, d_3, d_4\}$

Be as specific in your matches as possible

- ▶ You could use something simple like:

`\d+-\d+-\d+`

- ▶ But it will mistake a phone number for a SSN:

The screenshot shows a Splunk search interface. At the top, there's a section labeled "REGULAR EXPRESSION" with a text input field containing the regex `\d+-\d+-\d+`. Below this is a section labeled "TEST STRING" with two lines of text: "My phone number is 800-123-4567" and "My SSN is 123-45-6789". Both the phone number and the SSN are highlighted in blue, indicating they both match the regex. The background of the interface is light gray with blue accents.

A Better Match

- New regex:

`\d+-\d\d-\d+`

- Gets rid of Phone #'s, but what about other data?

REGULAR EXPRESSION

`\d+-\d\d-\d+`

TEST STRING

My phone number is 800-123-4567
My SSN is 123-45-6789
My birthday is 12-25-1960
I need part # 71-34-912

We're now so close

- ▶ This is exactly what a properly formatted SSN looks like:

`\d{3}-\d{2}-\d{4}`

- ▶ This defines a SSN, but it matches other things, too:

The screenshot shows a web interface for testing regular expressions. The 'REGULAR EXPRESSION' field contains the pattern `\d{3}-\d{2}-\d{4}`. The 'TEST STRING' field contains four lines of text: 'My phone number is 800-123-4567', 'My SSN is 123-45-6789', 'My birthday is 12-25-1960', and 'I need part # 8871-34-91268'. The matches for the SSN pattern are highlighted in blue: '123-45-6789' and '8871-34-91268'.

REGULAR EXPRESSION	TEST STRING
<code>\d{3}-\d{2}-\d{4}</code>	My phone number is 800-123-4567
	My SSN is 123-45-6789
	My birthday is 12-25-1960
	I need part # 8871-34-91268

987-65-4321

130 60.4 - [07/Jan 18:10:57:153] "GET /category.screen?category_id=GIFTS&SESSIONID=SD5L4FF10ADFF10 HTTP 1.1" 404 720 "http://buttercup-shopping.com/cart.do?action=view&itemId=EST-6&product_id=FJ-SW-G3...
128 241.220.82 - [07/Jan 18:10:57:123] "GET /product.screen?product_id=FL-DSH-01&SESSIONID=SD5L5FF6ADFF9 HTTP 1.1" 404 3322 "http://buttercup-shopping.com/cart.do?action=purchase&itemId=EST-3&product_id=K-CW-4...
131 27.160.0.0 - [07/Jan 18:10:56:156] "GET /oldlink?item_id=EST-26&SESSIONID=SD5L9FF1ADFF3 HTTP 1.1" 200 1318 "http://buttercup-shopping.com/changequantity&itemId=EST-6&SESSIONID=SD1B5L8FF-CB-0&SESSIONID=SD5L5FF6ADFF9 HTTP 1.1" 200 3865 GADT...
ows NT 5.1; SV1; .NET CLR 1.1.4322) "GET /cart.do?action=remove&itemId=EST-1&product_id=FLOWERS&SESSIONID=SD5L5FF6ADFF9 HTTP 1.1" 200 3865 GADT...
to:buttercup-shopping.com/RP-LI-022)" 468 125.17 14 10 1891 "GET /category.screen?category_id=SURPRISE&SESSIONID=SD5L5FF6ADFF9 HTTP 1.1" 200 1318 "http://buttercup-shopping.com/changequantity&itemId=EST-6&SESSIONID=SD1B5L8FF-CB-0&SESSIONID=SD5L5FF6ADFF9 HTTP 1.1" 200 3865 GADT...
opping.com/purchase&itemId=EST-3&product_id=K-CW-4" 200 1318 "http://buttercup-shopping.com/changequantity&itemId=EST-6&SESSIONID=SD1B5L8FF-CB-0&SESSIONID=SD5L5FF6ADFF9 HTTP 1.1" 200 3865 GADT...
0/buttercup-shopping.com/oldlink?item_id=EST-26&SESSIONID=SD5L9FF1ADFF3 HTTP 1.1" 200 1318 "http://buttercup-shopping.com/changequantity&itemId=EST-6&SESSIONID=SD1B5L8FF-CB-0&SESSIONID=SD5L5FF6ADFF9 HTTP 1.1" 200 3865 GADT...

- [illegible]

[illegible]

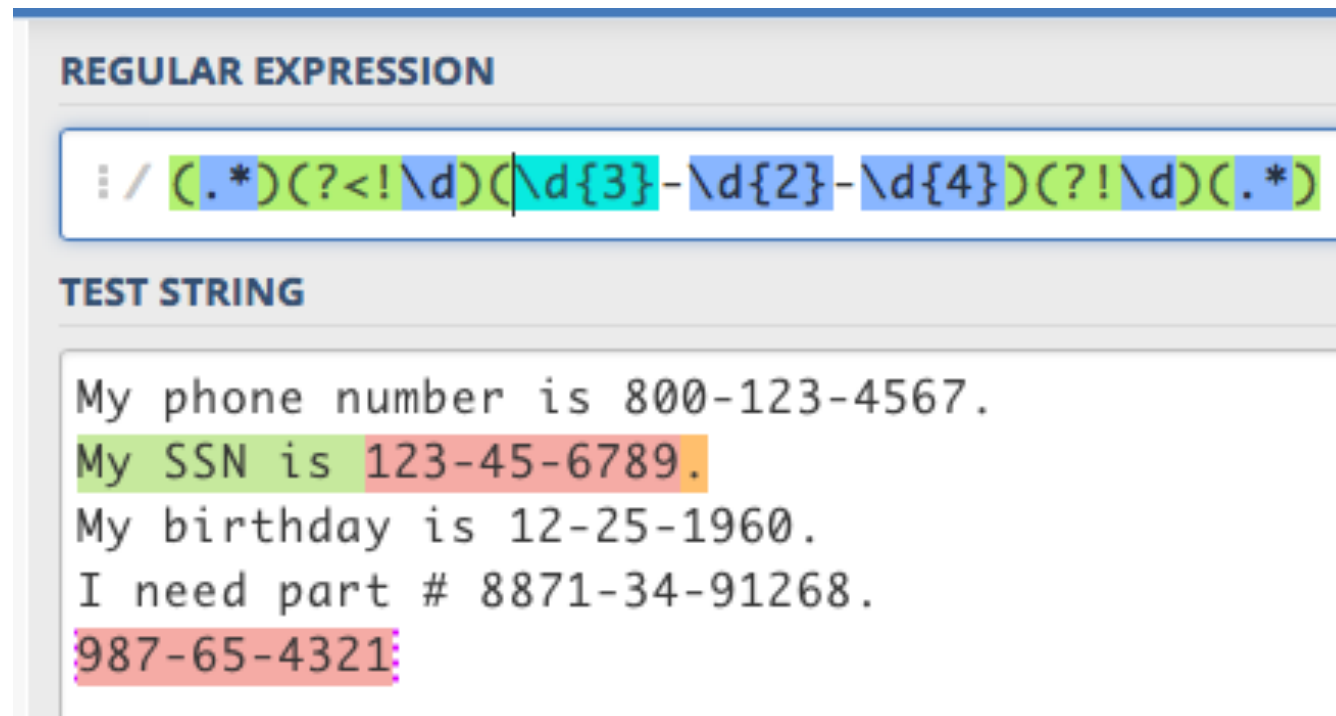
- 987-65-4321
- 130 60.4 - [07/Jan 18:10:57:153] "GET /category.screen?category_id=GIFTS&SESSIONID=SD5L4FF10ADFF10 HTTP 1.1" 404 720 "http://buttercup-shopping.com/cart.do?action=view&itemId=EST-6&product_id=F1-SW-G3...
128 241.220.82 - [07/Jan 18:10:57:123] "GET /product.screen?product_id=FL-DSH-01&SESSIONID=SD5L5FF6ADFF9 HTTP 1.1" 404 3322 "http://buttercup-shopping.com/cart.do?action=purchase&itemId=EST-3&product_id=GIFTS...
131 27.160.0.0 - [07/Jan 18:10:56:156] "GET /oldlink?item_id=EST-26&SESSIONID=SD5L9FF1ADFF3 HTTP 1.1" 200 1318 "http://buttercup-shopping.com/changequantity&itemId=EST-6&SESSIONID=SD1B5L8FF-CB-0A&SESSIONID=SD5L5FF6ADFF9 HTTP 1.1" 200 3865 GADT...
ows NT 5.1; SV1; .NET CLR 1.1.4322) "GET /oldlink?item_id=EST-26&SESSIONID=SD5L9FF1ADFF3 HTTP 1.1" 200 1318 "http://buttercup-shopping.com/changequantity&itemId=EST-6&SESSIONID=SD1B5L8FF-CB-0A&SESSIONID=SD5L5FF6ADFF9 HTTP 1.1" 200 3865 GADT...
to?action=shopping.com/rp-LI-022)" 468 125.17 14 10 189] "GET /cart.do?action=remove&itemId=EST-14...
opping.com/purchase&id=RP-LI-022)" 468 125.17 14 10 189] "GET /cart.do?action=remove&itemId=EST-14...
0/buttercup/case&i..."

[illegible]

So let's make it work in transforms.conf

- Grab the beginning and ending text of the event:

`(.*)((?<!\d)(\d{3}-\d{2}-\d{4})?(?!\\d))(.*)`



Conditions and Limitations

- ▶ This regex can't be done with SEDCMD in props.conf alone
 - The regex uses regex features not found in SED format
- ▶ Using a simple custom sourcetype, but it can be made a general transform
- ▶ Must capture all parts of the event.
- ▶ Will obfuscate only one SSN per event.

The props.conf

[testssn]

TRANSFORM-ssn = nossn

DATETIME_CONFIG = CURRENT

NO_BINARY_CHECK = true

SHOULD_LINEMERGE = false

disabled = false

The transforms.conf

[nossn]

REGEX=(?m)(.*)((?!\\d)(\\d{3}-\\d{2}-\\d{4})?(?!\\d)(.*))

FORMAT = \$1###-##-####\$3

DEST_KEY = _raw

New regex features shown

- ▶ **(?m)** – perform the regex on multi-line events
- ▶ **(?<!\d)** - Negative Lookbehind – *not preceeded by a digit* – no net character
- ▶ **(?! \d)** – Negative Lookahead – *not followed by a digit* – no net character

How the FORMAT works

- ▶ **FORMAT = \$1###-##-####\$3**
- ▶ **\$1** and **\$3** are capture group matches – from **(.*)** at beginning and end
- ▶ **\$2** is not used in the FORMAT, but it's the capture group for the SSN – from:
(\d{3}-\d{2}-\d{4})

After bringing in the data:

i	Time	Event
>	7/18/17 2:16:51.000 PM	###-###-#### host = Carys-MacBook-Pro-2.local source = ssn.log sourcetype = testssn
>	7/18/17 2:16:51.000 PM	I need part # 8871-34-91268. host = Carys-MacBook-Pro-2.local source = ssn.log sourcetype = testssn
>	7/18/17 2:16:51.000 PM	My birthday is 12-25-1960. host = Carys-MacBook-Pro-2.local source = ssn.log sourcetype = testssn
>	7/18/17 2:16:51.000 PM	My SSN is ###-###-####. host = Carys-MacBook-Pro-2.local source = ssn.log sourcetype = testssn
>	7/18/17 2:16:51.000 PM	My phone number is 800-123-4567. host = Carys-MacBook-Pro-2.local source = ssn.log sourcetype = testssn

It even will obfuscate the data import preview:

Source type: testssn

Save As

Event Breaks

Break Type

Auto

Every Line

Regex...

Timestamp

Extraction

Auto

Current time

Advanced...

Advanced

List

Format

20 Per Page

	Time	Event
1	7/18/17 2:15:56.000 PM	My phone number is 800-123-4567. timestamp = none
2	7/18/17 2:15:56.000 PM	My SSN is ###-###-####. timestamp = none
3	7/18/17 2:15:56.000 PM	My birthday is 12-25-1960. timestamp = none
4	7/18/17 2:15:56.000 PM	I need part # 8871-34-91268. timestamp = none
5	7/18/17 2:15:56.000 PM	###-###-#### timestamp = none



Greedy vs. Lazy Matches

⋮ / \d+6

12345678901234567890

:/ \d+?6

12345678901234567890

What is the difference?

Subtle difference, but big effect

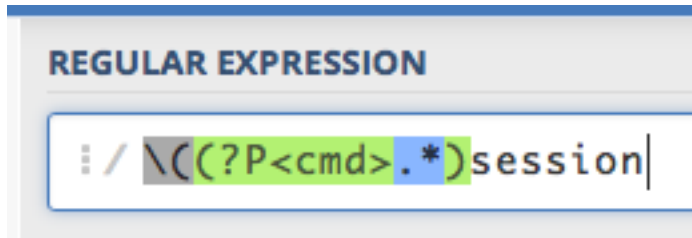
- ▶ **Greedy** – Grab as much as you can
- ▶ **Lazy** – Grab as little as you can
- ▶ The lazy match will continue only as far as it needs to, no further
 - `<.+?>` will match `<12345>`, while
 - `<.+>` will match both `<12345>` and `<12345><67890>`

SYNTAX: place a **?** After a ***** or **+**

The lazy match only goes to the first instance of a match following a multiple match

```
130.60.4 - - [07/Jun 18:10:57:153] "GET /category.screen?category_id=GIFTS&SESSIONID=5D5SLAFF10ADFF10 HTTP 1.1" 404 720 "http://buttercup-shopping.com/cart.do?action=view&itemId=EST-6&product_id=FI-SW-03"
128.241.220.82 - - [07/Jun 18:10:57:123] "GET /product.screen?product_id=FL-DSH-01&SESSIONID=5D5SL7FF6ADFF9 HTTP 1.1" 404 3322 "http://buttercup-shopping.com/cart.do?action=purchase&itemId=EST-26&product_id=K9-CB-01"
ows NT 5.1; SV1: - - [07/Jun 18:10:56:156] "GET /oldlink?item_id=EST-26&SESSIONID=5D5SL9FF1ADFF3 HTTP 1.1" 200 1318 "http://buttercup-shopping.com/cart.do?action=changequantity&itemId=EST-18&product_id=AV-CB-01&SESSIONID=5D5SL8FF3ADFF9 HTTP 1.1" 200 2423 "http://buttercup-shopping.com/cart.do?action=purchase&itemId=EST-26&product_id=K9-CB-01"
item_id=EST-16&product_id=RP-LI-02" 468 125.17 14.000000000000000 "GET /category.screen?category_id=FLOWERS&SESSIONID=5D5SL8FF1ADFF3 HTTP 1.1" 200 3885 "http://buttercup-shopping.com/cart.do?action=purchase&itemId=EST-26&product_id=K9-CB-01"
toaction=purchase&itemId=EST-26&product_id=K9-CB-01" 468 125.17 14.000000000000000 "GET /category.screen?category_id=FLOWERS&SESSIONID=5D5SL8FF1ADFF3 HTTP 1.1" 200 3885 "http://buttercup-shopping.com/cart.do?action=purchase&itemId=EST-26&product_id=K9-CB-01"
opping.com/purchase&itemId=EST-26&product_id=K9-CB-01" 468 125.17 14.000000000000000 "GET /category.screen?category_id=FLOWERS&SESSIONID=5D5SL8FF1ADFF3 HTTP 1.1" 200 3885 "http://buttercup-shopping.com/cart.do?action=purchase&itemId=EST-26&product_id=K9-CB-01"
```

Greedy Example



```
led password for invalid user geo
unix(sshd:session): session open
led password for invalid user yp
led password for invalid user ema
led password for invalid user loc
led password for invalid user mys
im_unix(sshd:session): session ope
led password for myuan from 10.1.
```

```
130.60.4 - - [07/Jan 18:10:57:153] "GET /category.screen?category_id=GIFTS&SESSIONID=5D5SLAFF10ADFF10 HTTP 1.1" 404 720 "http://buttercup-shopping.com/cart.do?action=view&itemId=EST-6&product_id=F1-SW-03"
128.241.220.82 - - [07/Jan 18:10:57:123] "GET /product.screen?product_id=FL-DSH-01&SESSIONID=5D5SL7FF6ADFF9 HTTP 1.1" 404 3322 "http://buttercup-shopping.com/cart.do?action=purchase&itemId=EST-26&product_id=K9-CW-01"
317 27.160.0.0 - - [07/Jan 18:10:56:156] "GET /oldlink?item_id=EST-26&SESSIONID=5D5SL9FF1ADFF3 HTTP 1.1" 200 1318 "http://buttercup-shopping.com/cart.do?action=changequantity&itemId=EST-18&product_id=AV-CB-01&SESSIONID=5D185LBFF3ADFF9"
item_id=EST-16&product_id=RP-LI-02" 468 125.17 14.11.189) "GET /category.screen?category_id=FLOWERS&SESSIONID=5D5SL7FF6ADFF9 HTTP 1.1" 200 3885 "http://buttercup-shopping.com/cart.do?action=purchase&itemId=EST-26&product_id=K9-CW-01"
action=purchase&itemId=EST-26&product_id=K9-CW-01" 468 125.17 14.11.189) "GET /category.screen?category_id=FLOWERS&SESSIONID=5D5SL7FF6ADFF9 HTTP 1.1" 200 3885 "http://buttercup-shopping.com/cart.do?action=purchase&itemId=EST-26&product_id=K9-CW-01"
action=purchase&itemId=EST-26&product_id=K9-CW-01" 468 125.17 14.11.189) "GET /category.screen?category_id=FLOWERS&SESSIONID=5D5SL7FF6ADFF9 HTTP 1.1" 200 3885 "http://buttercup-shopping.com/cart.do?action=purchase&itemId=EST-26&product_id=K9-CW-01"
```

REGULAR EXPRESSION

```
ed password for invalid user ge
unix(sshd:session): session op
ed password for invalid user yp
ed password for invalid user en
ed password for invalid user lc
ed password for invalid user my
_unix(sshd:session): session op
ed password for myuan from 10.1
```

Second Look - Greedy

REGULAR EXPRESSION

1 match, 689 steps (~31ms)

From `(?P<start>.+)` to `(?P<end>.+):` \s

/g

TEST STRING

SWITCH TO UNIT TESTS ▶

Number of AuthenticationFailed events exceeds configured threshold during configured interval of time 1 within 3 minutes on cluster StandAloneCluster. There are 2 AuthenticationFailed events (up to 30) received during the monitoring interval From Wed Aug 03 10:25:00 PHT 2016 to Wed Aug 03 10:28:00 PHT 2016: TimeStamp : 8/3/16 10:26 AM LoginFrom : 172.12.34.40 Interface : VMREST UserID : JacobMD AppID : Cisco Tomcat ClusterID : NodeID : APPHMANAOVM001 TimeStamp : Wed Aug 03 10:26:13 PHT 2016 TimeStam::Status=2,cleared

Second Look - Lazy

REGULAR EXPRESSION

1 match, 1016 steps (~8ms)

/ From (?P<start>.+) to (?P<end>.+?):\s

/g

TEST STRING

SWITCH TO UNIT TESTS

Number of AuthenticationFailed events exceeds configured threshold during configured interval of time 1 within 3 minutes on cluster StandAloneCluster. There are 2 AuthenticationFailed events (up to 30) received during the monitoring interval From Wed Aug 03 10:25:00 PHT 2016 to Wed Aug 03 10:28:00 PHT 2016: TimeStamp : 8/3/16 10:26 AM LoginFrom : 172.12.34.40 Interface : VMREST UserID : JacobMD AppID : Cisco Tomcat ClusterID : NodeID : APPHMANAOVM001 TimeStamp : Wed Aug 03 10:26:13 PHT 2016 TimeStam::Status=2,cleared

Choose Wisely

- ▶ Greedy may cross long segments
- ▶ Lazy may stop prematurely
- ▶ **Try it on various data sets** to make sure it will do what you want



Embedding

Multiple field extractions from one piece of data

► Problem:

- Extract two different fields from the **exact same piece of data**
- Only want to use **one regex** – for efficiency if nothing else
- Need both the **Domain** only and the **whole URL** from an access log

Source and Results

► Data:

- 1501408932.060 16922 108.65.113.83 TCP_REFRESH_HIT/200 474 GET <http://damtare.by.ru/id.txt> myuan@buttercupgames.com DIRECT/damtare.by.ru text/html DEFAULT_CASE-DefaultGroup-Demo_Clients-NONE-NONE-DefaultRouting <IW_scty,-6.9,0,-,-,-,0,-,-,-,-,-,IW_scty,-> - -

► Desired Fields:

- Domain: damtare.by.ru
- URL: <http://damtare.by.ru/id.txt>

Regex

REGULAR EXPRESSION

28 matches, 439 steps (~26ms)

```
/(?P<URL>http:\\\\(?P<domain>[^\\/]+)\S+)
```

/g |

TEST STRING

SWITCH TO UNIT TESTS ▶

```
1501400746.799 563 89.11.192.18 TCP_MISS/403 4206 GET
http://www.hybridarcade.com/ pbunch@buttercupgames.com
DIRECT/www.hybridarcade.com text/html DEFAULT_CASE-
DefaultGroup-Demo_Clients-NONE-NONE-DefaultRouting
<IW_game,ns,0,-,-,-,0,-,-,-,-,-,IW_game,-> - -
1501401212.060 16922 173.192.201.242 TCP_REFRESH_HIT/200 474
GET http://damtare.by.ru/id.txt pcallahan@buttercupgames.com
DIRECT/damtare.by.ru text/html DEFAULT_CASE-DefaultGroup-
Demo_Clients-NONE-NONE-DefaultRouting <IW_scty,-6.9,0,-,-,-,
,0,-,-,-,-,-,IW_scty,-> - -
1501402224.788 931 112.111.162.4 TCP_REFRESH_HIT/200 3227 GET
http://www.goppo.com/ acurry@buttercupgames.com
DIRECT/www.goppo.com text/html DEFAULT_CASE-DefaultGroup-
```


Regex101.com vs Splunk

- Slashes need escaping in regex101, but not in Splunk:

(?P<URL>http:W(?P<domain>[^V]+)\S+)

VS

(?P<URL>http://(?P<domain>[^/]+\S+)

What? ... Oh, now I see.

(?P<URL>http://(?P<domain>[^/]+)\S+)

^

^

^

^

130.60.4 - - [07/Jan 18:10:57:153] "GET /category.screen?category_id=GIFTS&SESSIONID=5D5SLAFF10ADFF10 HTTP 1.1" 404 720 "http://buttercup-shopping.com/cart.do?action=view&itemId=EST-6&product_id=FI-SW-03"
128.241.220.82 - - [07/Jan 18:10:57:123] "GET /product.screen?product_id=FL-DSH-01&SESSIONID=5D5SL7FF6ADFF0 HTTP 1.1" 404 3322 "http://buttercup-shopping.com/category.screen?category_id=GIFTS"
317 27.160.0.0 - - [07/Jan 18:10:56:156] "GET /oldlink?item_id=EST-26&SESSIONID=5D5SL9FF1ADFF3 HTTP 1.1" 200 1318 "http://buttercup-shopping.com/cart.do?action=purchase&itemId=EST-26&product_id=FI-SW-03"
125.17 14.14.14.14 - - [07/Jan 18:10:55:187] "GET /category.screen?category_id=FLOWERS&SESSIONID=5D5SL8FF1ADFF3 HTTP 1.1" 200 3885 "http://buttercup-shopping.com/cart.do?action=purchase&itemId=EST-26&product_id=FI-SW-03"
125.17 14.14.14.14 - - [07/Jan 18:10:55:188] "GET /category.screen?category_id=FLOWERS&SESSIONID=5D5SL8FF1ADFF3 HTTP 1.1" 200 3885 "http://buttercup-shopping.com/cart.do?action=purchase&itemId=EST-26&product_id=FI-SW-03"

Results in Splunk

You can't do this in the FET without doing your own regex!

Regular Expression

[Regular Expression Reference](#) [View in Search](#)

(?P<URL>http://(?P<domain>[^/]+\S+)

Preview

Save

Events

URL

domain

✓ 30 events (before 8/31/17 4:37:35.000 PM)

Original search included: ☒

20 per page

< Prev

1

2

Next >

filter

Apply

Sample: 1,000 events

All events

All Events

Matches

Non-Matches

	_raw	URL	domain
✓		http://damtare.by.ru/id.txt	damtare.by.ru
✓		http://www.lowermybills.com/images/home_banner/home_auto_insurance.jpg	www.lowermybills.com
✓		http://www.lowermybills.com/dwr/engine.js	www.lowermybills.com
✓		http://www.lowermybills.com/dwr/interface/RemoteLogger.js	www.lowermybills.com
✓		http://www.lowermybills.com/	www.lowermybills.com
✓		http://damtare.by.ru/id.txt	damtare.by.ru

splunk>

.conf2017



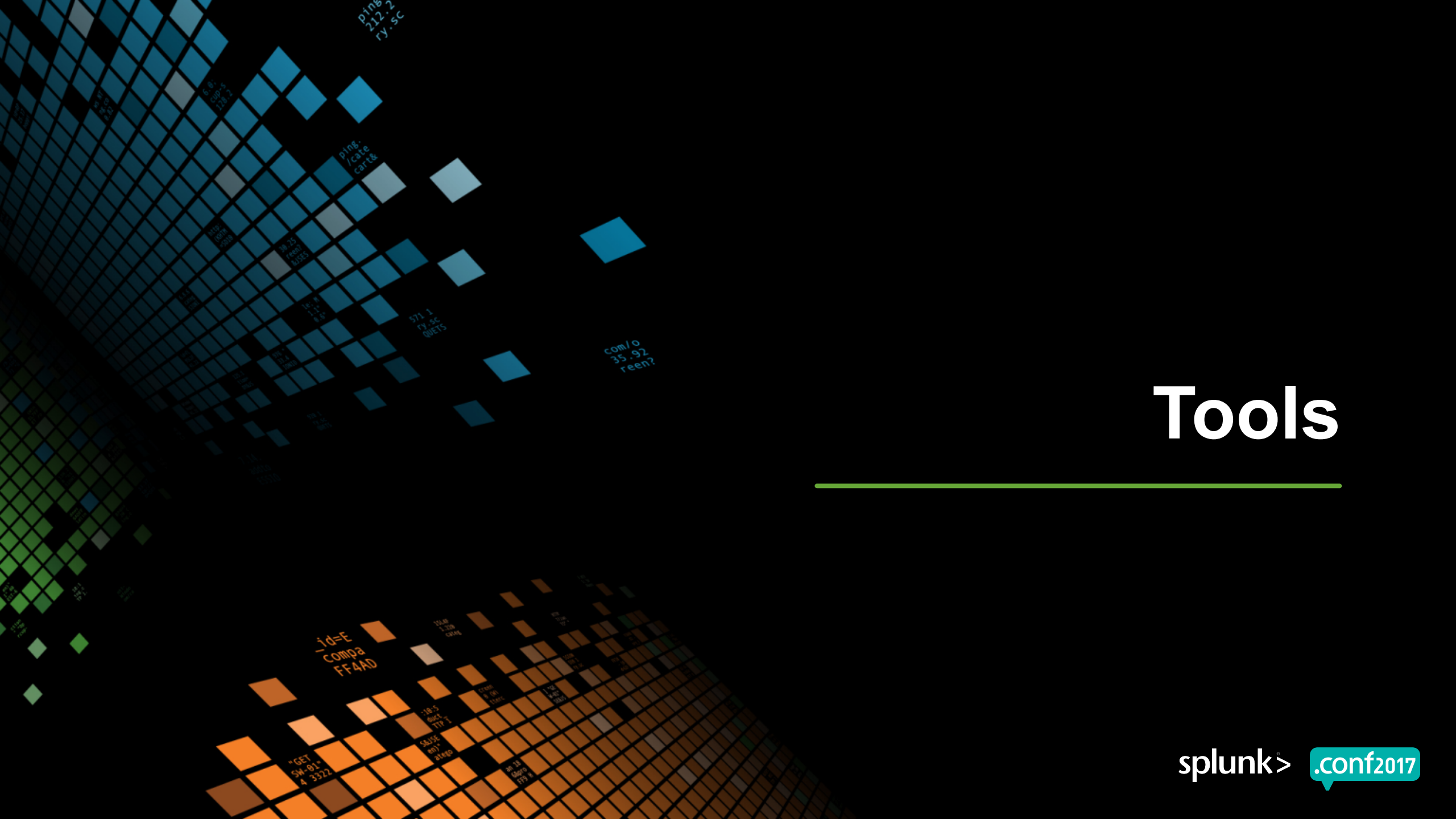
Other Notes

Performance Considerations

- ▶ Some complex field extractions can be costly
- ▶ Some complex regular expressions can be costly
- ▶ **Use the Job Inspector** to see if there is a difference in doing on complex field extraction vs man simple field extractions (| rex | **vs** | rex | rex | rex)
- ▶ **Sometimes the readability is more important** than the performance

Maintenance Considerations

- ▶ The **complex field extractions** (for example, one that extracts 6 fields at once) **may be easier to maintain** than multiple simple extractions (where you would have 6 different fields extracted by 6 different regexes)
- ▶ Your own field extractions will probably be easier to maintain than those created by the Field Extraction Tool – just **write your own regexes better** than the FET
- ▶ **Regexes can save you a lot of headaches compared to using non-regex field extractions** (one user was trying to extract data using 200 non-regex evals compared to 6 regexes that accomplished the same thing!)



Tools

Regex101 Web Page

<http://regex101.com>

regular expressions 101

@regex101 donate contact bug reports & feedback wiki

SAVE & SHARE

save regex %s

FLAVOR

pcre (php) ✓

javascript

python

golang

TOOLS

code generator

regex debugger

REGULAR EXPRESSION

47 matches, 23269 steps (~22ms)

TEST STRING

SWITCH TO UNIT TESTS

EXPLANATION

▼ / (for (invalid user)(user))?(sudo:)(P<user>\S+) (from|by)? /g

▼ 1st Capturing Group
(for (invalid user)(user))?(sudo:)

▼ 1st Alternative
(for (invalid user)(user))

▼ 2nd Capturing Group
(for (invalid user)(user))

for matches the characters for literally (case sensitive)

▼ 3rd Capturing Group
(invalid user)(user)

? Quantifier — Matches between zero and one times, as many times as possible, giving back as needed (greedy)

MATCH INFORMATION

Match 1

Full match 58-71 `for root from`

Group 1. 58-62 `for `

Group 2. 58-62 `for `

Group `user` 62-66 `root`

Group 8. 67-71 `from`

Match 2

Full match 160-195 `for invalid user administrator from`

Group 1. 160-177 `for invalid user `

Group 2. 160-177 `for invalid user `

Group 3. 164-177 `invalid user `

QUICK REFERENCE

Search reference

all tokens

common tokens ✓

general tokens

anchors

meta sequences

quantifiers

group constructs

character classes

a single character of: a, b or c [abc]

a character except: a, b or c [^abc]

a character in the range: a-z [a-z]

a character not in the ran... [^a-z]

a character in the rang... [a-zA-Z]

any single character .

any whitespace character \s

any non-whitespace character \S

any digit \d

Regexr Web Page

<http://regexr.com> - Doesn't do PCRE!!

RegExrv2.1

by gskinner RegExrv1 GitHub Tutorial

Library

Help

Reference

Cheatsheet

Examples

Community

Favourites

RegExr is an online tool to learn, build, & test Regular Expressions (RegEx / RegExp).

- Results update in **real-time** as you type.
- Roll over** a match or expression for details.
- Save & share** expressions with others.
- Use **Tools** to explore your results.
- Browse the **Library** for help & examples.
- Undo & Redo** with Cmd-Z / Y.
- Search for & rate **Community** patterns.

Expression

share save flags

`/((for ((invalid user)|(user)))|(sudo: ))(\S+) (from|by)?/g`

47 matches

Text

```
Mon Jun 06 2016 11:04:54 www1 sudo: nsharpe ; TTY=pts/0 ; PWD=/home/nsharpe ; USER=root ; COMMAND=/bin/su
Mon Jun 06 2016 11:04:54 www1 sshd[2572]: Failed password for ncscd from 10.2.10.163 port 3894 ssh2
Mon Jun 06 2016 11:05:20 www1 sshd[3053]: Failed password for invalid user administrator from 10.2.10.163 port 1662
ssh2
Mon Jun 06 2016 11:06:12 www1 sshd[5350]: Failed password for invalid user administrator from 10.2.10.163 port 3666
ssh2
Mon Jun 06 2016 11:06:44 www1 sshd[4115]: Failed password for invalid user administrator from 10.2.10.163 port 3568
ssh2
Mon Jun 06 2016 11:07:24 www1 sshd[22652]: Accepted password for djohnson from 10.3.10.46 port 9128 ssh2
Mon Jun 06 2016 11:07:24 www1 sshd[5331]: Failed password for invalid user administrator from 10.2.10.163 port 4762
ssh2
```

Tools

Replace

List

Details

Explain

\$7\n

root

administrator

george

djohnson

yp

email

local

mysql

djohnson

myuan

vn

Improve Regex Performance

► <https://www.loggly.com/blog/five-invaluable-techniques-to-improve-regex-performance/>

Try your regex prowess

► Regex Golf

- <https://alf.nu/RegexGolf>
- <https://www.oreilly.com/learning/regex-golf-with-peter-norvig>

► Regex Crosswords

- <https://regexcrossword.com>
- <https://mariolurig.com/crossword/>

Splunk Answers and Docs

Learn from others – ask questions – get answers

► <http://answers.splunk.com/>

Splunk Documentation

► <https://docs.splunk.com/Documentation/Splunk/6.4.3/Knowledge/AboutSplunkregularexpressions>

► Splunk **regex** Slack channel

130.60.4 - - [07/Jan 18:10:57:153] "GET /category.screen?category_id=GIFTS&JSESSIONID=5D5SLAFF10ADFF10 HTTP 1.1" 404 720 "http://buttercup-shopping.com/cart.do?action=view&itemId=EST-6&product_id=F1-SW-03"
128.241.220.82 - - [07/Jan 18:10:57:123] "GET /product.screen?product_id=FL-DSH-01&JSESSIONID=5D5SL7FF6ADFF0 HTTP 1.1" 200 1318 "http://buttercup-shopping.com/cart.do?action=purchase&itemId=EST-26&product_id=FL-DSH-01"
317 27.160.0.0 - - [07/Jan 18:10:56:156] "GET /oldlink?item_id=EST-26&JSESSIONID=5D5SL9FF1ADFF3 HTTP 1.1" 200 1318 "http://buttercup-shopping.com/cart.do?action=changequantity&itemId=EST-18&product_id=AV-CB-01&JSESSIONID=5D5SL8FF3ADFF9 HTTP 1.1" 200 2423 "http://buttercup-shopping.com/cart.do?action=purchase&itemId=EST-26&product_id=FL-DSH-01"
128.241.220.82 - - [07/Jan 18:10:57:123] "GET /category.screen?category_id=GIFTS&JSESSIONID=5D5SLAFF10ADFF10 HTTP 1.1" 404 720 "http://buttercup-shopping.com/cart.do?action=view&itemId=EST-6&product_id=F1-SW-03"
128.241.220.82 - - [07/Jan 18:10:57:123] "GET /product.screen?product_id=FL-DSH-01&JSESSIONID=5D5SL7FF6ADFF0 HTTP 1.1" 200 1318 "http://buttercup-shopping.com/cart.do?action=purchase&itemId=EST-26&product_id=FL-DSH-01"
317 27.160.0.0 - - [07/Jan 18:10:56:156] "GET /oldlink?item_id=EST-26&JSESSIONID=5D5SL9FF1ADFF3 HTTP 1.1" 200 1318 "http://buttercup-shopping.com/cart.do?action=changequantity&itemId=EST-18&product_id=AV-CB-01&JSESSIONID=5D5SL8FF3ADFF9 HTTP 1.1" 200 2423 "http://buttercup-shopping.com/cart.do?action=purchase&itemId=EST-26&product_id=FL-DSH-01"
128.241.220.82 - - [07/Jan 18:10:57:123] "GET /category.screen?category_id=GIFTS&JSESSIONID=5D5SLAFF10ADFF10 HTTP 1.1" 404 720 "http://buttercup-shopping.com/cart.do?action=view&itemId=EST-6&product_id=F1-SW-03"
128.241.220.82 - - [07/Jan 18:10:57:123] "GET /product.screen?product_id=FL-DSH-01&JSESSIONID=5D5SL7FF6ADFF0 HTTP 1.1" 200 1318 "http://buttercup-shopping.com/cart.do?action=purchase&itemId=EST-26&product_id=FL-DSH-01"
317 27.160.0.0 - - [07/Jan 18:10:56:156] "GET /oldlink?item_id=EST-26&JSESSIONID=5D5SL9FF1ADFF3 HTTP 1.1" 200 1318 "http://buttercup-shopping.com/cart.do?action=changequantity&itemId=EST-18&product_id=AV-CB-01&JSESSIONID=5D5SL8FF3ADFF9 HTTP 1.1" 200 2423 "http://buttercup-shopping.com/cart.do?action=purchase&itemId=EST-26&product_id=FL-DSH-01"

Acknowledgements

► Lisa Guinn

- Inspiration and guidance in preparing
- Data set to use in examples

130.60.4 - - [07/Jun 18:10:57:153] "GET /category.screen?category_id=GIFTS&JSESSIONID=5D5SLAFF10ADFF10 HTTP 1.1" 404 720 "http://buttercup-shopping.com/cart.do?action=view&itemId=EST-6&product_id=FI-SW-03"
128.241.220.82 - - [07/Jun 18:10:57:123] "GET /product.screen?product_id=FL-DSH-01&JSESSIONID=5D5SL7FF6ADFF0 HTTP 1.1" 404 3322 "http://buttercup-shopping.com/cart.do?action=purchase&itemId=EST-26&product_id=FI-SW-03"
317 27.160.0.0 - - [07/Jun 18:10:56:156] "GET /oldlink?item_id=EST-26&JSESSIONID=5D5SL9FF1ADFF3 HTTP 1.1" 200 1318 "http://buttercup-shopping.com/cart.do?action=changequantity&itemId=EST-18&product_id=AV-CB-01&JSESSIONID=5D5SL8FF3ADFF9 HTTP 1.1" 200 2423 "http://buttercup-shopping.com/cart.do?action=purchase&itemId=EST-26&product_id=FI-SW-03"
130.60.4 - - [07/Jun 18:10:57:153] "GET /category.screen?category_id=GIFTS&JSESSIONID=5D5SLAFF10ADFF10 HTTP 1.1" 404 720 "http://buttercup-shopping.com/cart.do?action=view&itemId=EST-6&product_id=FI-SW-03"
128.241.220.82 - - [07/Jun 18:10:57:123] "GET /product.screen?product_id=FL-DSH-01&JSESSIONID=5D5SL7FF6ADFF0 HTTP 1.1" 404 3322 "http://buttercup-shopping.com/cart.do?action=purchase&itemId=EST-26&product_id=FI-SW-03"
317 27.160.0.0 - - [07/Jun 18:10:56:156] "GET /oldlink?item_id=EST-26&JSESSIONID=5D5SL9FF1ADFF3 HTTP 1.1" 200 1318 "http://buttercup-shopping.com/cart.do?action=changequantity&itemId=EST-18&product_id=AV-CB-01&JSESSIONID=5D5SL8FF3ADFF9 HTTP 1.1" 200 2423 "http://buttercup-shopping.com/cart.do?action=purchase&itemId=EST-26&product_id=FI-SW-03"
130.60.4 - - [07/Jun 18:10:57:153] "GET /category.screen?category_id=GIFTS&JSESSIONID=5D5SLAFF10ADFF10 HTTP 1.1" 404 720 "http://buttercup-shopping.com/cart.do?action=view&itemId=EST-6&product_id=FI-SW-03"
128.241.220.82 - - [07/Jun 18:10:57:123] "GET /product.screen?product_id=FL-DSH-01&JSESSIONID=5D5SL7FF6ADFF0 HTTP 1.1" 404 3322 "http://buttercup-shopping.com/cart.do?action=purchase&itemId=EST-26&product_id=FI-SW-03"
317 27.160.0.0 - - [07/Jun 18:10:56:156] "GET /oldlink?item_id=EST-26&JSESSIONID=5D5SL9FF1ADFF3 HTTP 1.1" 200 1318 "http://buttercup-shopping.com/cart.do?action=changequantity&itemId=EST-18&product_id=AV-CB-01&JSESSIONID=5D5SL8FF3ADFF9 HTTP 1.1" 200 2423 "http://buttercup-shopping.com/cart.do?action=purchase&itemId=EST-26&product_id=FI-SW-03"

THE CHURCH OF
JESUS CHRIST
OF LATTER-DAY SAINTS



Questions?
