Tricks You Can Do with NetFlow Analytics

How to extract value from this ubiquitous data source

David J. Cavuto, CISSP | Staff Sales Engineer, NY Metro

October 2018 | Version 2.0

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. Faor important factors that may cause actual results to differ from those contained in our forward-looking statements, please review our filings with the SEC.

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

splunk> .conf18

Bio

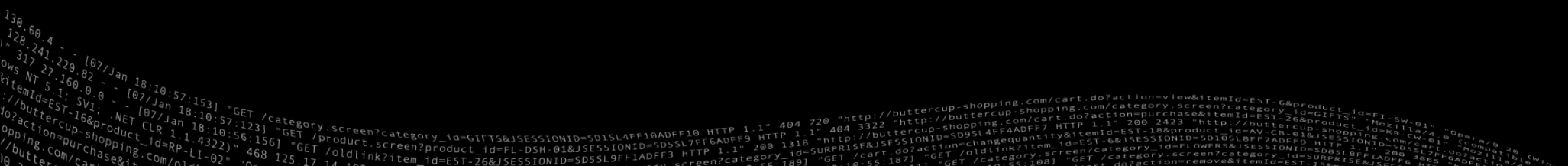
David J. Cavuto

- Bell Labs
  - Principal Engineer - Lucent VPN Firewall
- AT&T
  - Network security and analytics
- Narus
  - Product Manager – Narus Cyber Analytics
- Splunk
  - Sales Engineer, Security SME
  - Principal Product Manager – Splunk App for Stream
  - Principal Product Manager – Data Ecosystem Area

- David J. Cavuto
  dcavuto@splunk.com

Agenda

- NetFlow Introduction
  - Format and structure
  - Different Names and Versions
  - Netflow and Splunk Stream
- Basic Analytics
- IToA Analytics
- Security Analytics
- Wrap Up



splunk> .conf18

NetFlow Introduction

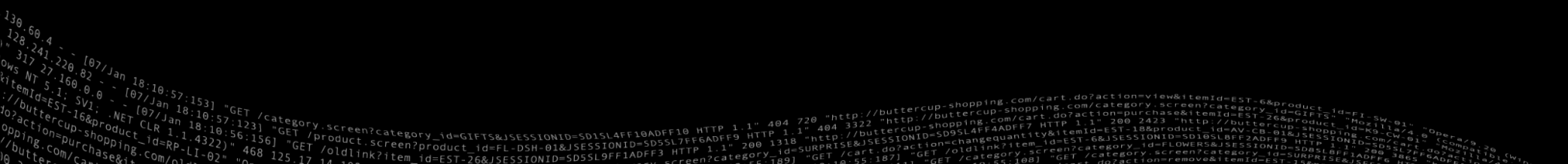
Background and Technology Overview

splunk> .conf18

NetFlow Inspiration

Cisco originated it to gain visibility into router traffic

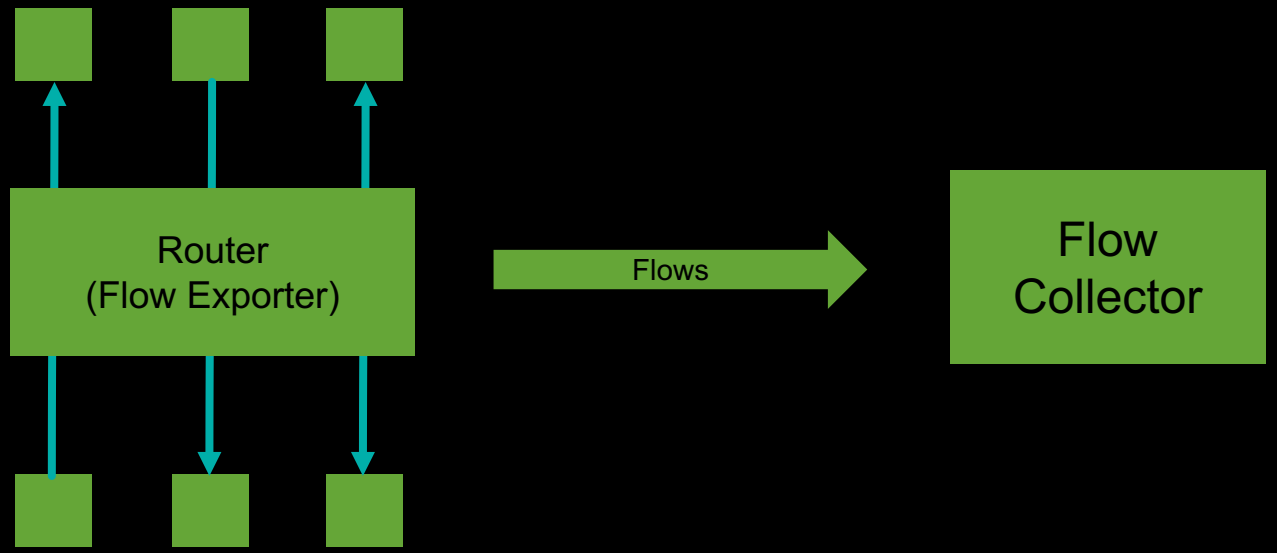
- Designed to be used on Routers
- Stateful, but minimizing memory usage
- Unidirectional
  - All TCP and some UDP generate TWO independent flow records
  - Often difficult to correlate deterministically
- New Versions added more features later



NetFlow Architecture

Two Main Components

- Traffic is observed via Flow Exporter
  - Can be inband (eg router or switch) or out of band (eg packet capture)
- Flow Exporter sends to Flow Collector



NetFlow Format and Structure

Version 5

Bytes	Contents	Description
0-3	srcaddr	Source IP address
4-7	dstaddr	Destination IP address
8-11	nexthop	IP address of next hop router
12-13	input	SNMP index of input interface
14-15	output	SNMP index of output interface
16-19	dPkts	Packets in the flow
20-23	dOctets	Total number of Layer 3 bytes in the packets of the flow
24-27	first	SysUptime at start of flow
28-31	last	SysUptime at the time the last packet of the flow was received
32-33	srcport	TCP/UDP source port number or equivalent
34-35	dstport	TCP/UDP destination port number or equivalent
36	pad1	Unused (zero) bytes
37	tcp_flags	Cumulative OR of TCP flags
38	prot	IP protocol type (for example, TCP = 6; UDP = 17)
39	tos	IP type of service (ToS)
40-41	src_as	Autonomous system number of the source, either origin or peer
42-43	dst_as	Autonomous system number of the destination, either origin or peer
44	src_mask	Source address prefix mask bits
45	dst_mask	Destination address prefix mask bits
46-47	pad2	Unused (zero) bytes

- Most Common version
- Available on just about every device that generates Flow messages
- Has flow metadata and statistics information
- Note that flow is UNIDIRECTIONAL
  - Each metric (packets, bytes, etc) only has a single value

Source: plixer.com

splunk> .conf18

Versions and Names

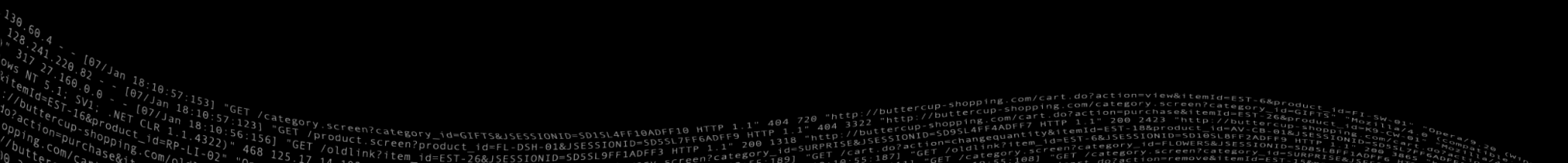
At least 3 primary versions in popular use

Flow Versions:

- NetFlow v5
- NetFlow v9
  - Templates
  - Vendor Extensions
- IPFIX (v10)
- sFlow - switches

Other Names:

- jFlow
- cFlowd



Splunk Stream and NetFlow

Splunk Stream is a NetFlow Collector!

- Stream can act as an active NetFlow collector (not promiscuous capture)
- Flow v5, v9, IPFIX
- Templates and Vendor Extensions
- 450,000 flows/second
  - (32 core / 64 GB in Independent Stream Forwarder mode)

splunk> .conf18

Basic Analytics

Basic NetFlow Analytics

Internal Host Inventory

Shows which hosts are inside your network

- Determine internal networks
  - Can use macro with lookup table if complex
  - Here we use cidrmatch for simple RFC1918 internal networks
- Summarize with stats
- Optionally output with outputcsv to store and diff

Search Syntax

```
sourcetype=stream:netflow
| where (cidrmatch("192.168.1.0/24", src_ip)
OR cidrmatch("10.0.0.0/8", src_ip)
OR cidrmatch("172.16.0.0/16", src_ip))
| stats count by src_ip
| sort +src_ip
```

src_ip	count
10.0.0.4	403
10.100.0.97	997
192.168.1.1	65834
192.168.1.2	21089
192.168.1.3	15070
192.168.1.5	7845
192.168.1.7	5562
192.168.1.9	26989
192.168.1.10	63
192.168.1.11	83
192.168.1.12	255
192.168.1.33	373
192.168.1.34	169
192.168.1.35	451
192.168.1.36	8
192.168.1.37	443
192.168.1.38	189
192.168.1.39	9
192.168.1.40	6579
192.168.1.41	1406

Visualization

Session Stitching

Connects the two sides of the unidirectional flow records

- Use “Client” and “Server” to differentiate different roles of “src” and “dest”
- Transaction is used to stitch two related flow records using client/server addresses and ports
- Here we use HTTP/S to nail down a server port

Search Syntax

```
sourcetype=netflow
| where src_port in (80,443) OR dest_port in (80,443)
| eval direction=if(dest_port in (80,443), "forward", "reverse")
| eval client_ip=if(direction=="forward", src_ip, dest_ip)
| eval server_ip=if(direction=="reverse", src_ip, dest_ip)
| eval client_port=if(direction=="forward", src_port, dest_port)
| eval server_port=if(direction=="reverse", src_port, dest_port)
| eval cs_bytes=if(direction=="forward",bytes, cs_bytes)
| eval sc_bytes=if(direction=="reverse",bytes, sc_bytes)
| transaction maxpause=10s maxevents=2
startswith=direction="forward" endswith=direction="reverse"
client_ip server_ip client_port server_port
| where eventcount=2
```

_time	client_ip	client_port	server_ip	server_port	direction	cs_bytes	sc_bytes
7/22/17 4:58:02.036 AM	91.230.47.3	55073	192.168.1.5	80	forward reverse	305	727
7/22/17 3:22:52.228 AM	192.168.1.101	61144	64.4.27.50	443	forward reverse	2004	5161
7/22/17 3:22:52.228 AM	192.168.1.101	61145	64.4.27.50	443	forward reverse	2828	5113
7/22/17 3:22:52.228 AM	192.168.1.101	61147	65.36.0.144	80	forward reverse	516	534
7/22/17 2:37:46.119 AM	211.22.90.247	23373	192.168.1.5	80	forward reverse	424	797
	.168.1.40	48664	194.186.47.19	80	forward reverse	523	2759
	.168.1.101	60971	157.55.240.126	443	forward reverse	2828	5113
	.168.1.101	60970	157.55.240.126	443	forward reverse	2050	5177
	230.47.3	43247	192.168.1.5	80	forward reverse	305	727

Visualization

splunk> .conf18

ITOA Analytics

Bandwidth and Host Monitoring

splunk> .conf18

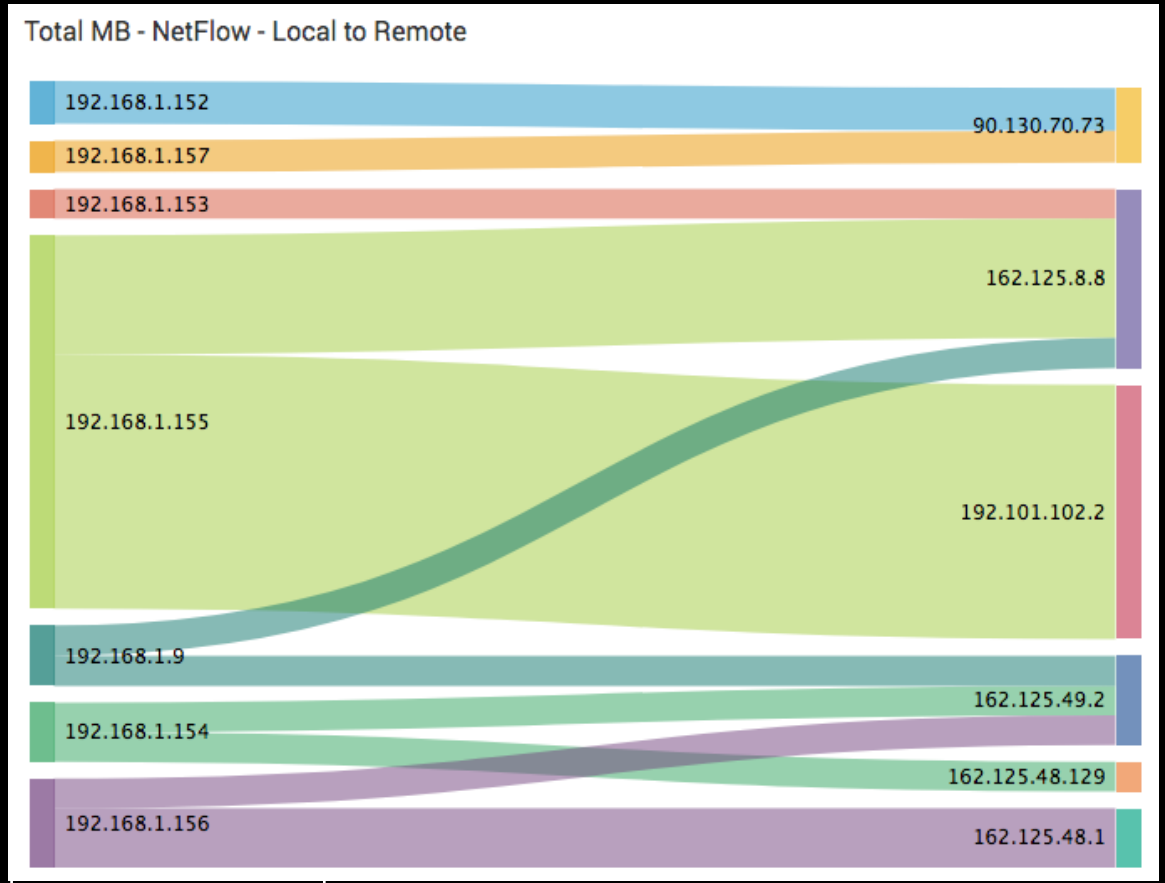
Bandwidth per Host

Shows how much bandwidth each host is using

- Determine host-host connectivity in total bytes or bits/second
- Use the “bytes” field and “stats” command to perform arithmetic sum
- Group by source and destination for summary
- Add an optional threshold to filter out noise
- Visualize with Sankey diagram (downloadable)

Search Syntax

```
sourcetype=netflow
| rename sum(bytes) AS sumbytes
| eval mb=round(sumbytes/(1024*1024),1)
| stats sum(mb) AS total_mb by src_ip,
dest_ip
| where total_mb>xxx
```



Visualization

splunk> .conf18

Internal Server Ports

Shows which hosts are acting as servers

- Select internal destination hosts
  - `is_internal_net` macro, matching previous expression
- Use heuristic: src_port > dest_port
- Display ports and counts for each IP
- Especially useful with sFlow (from internal switches)

Search Syntax

```
sourcetype=netflow
| `is_internal_net(dest_ip)`
| where src_port>=dest_port AND dest_port!=0
| stats count by dest_port,dest_ip
| eval dpc=dest_port." (".count.")"
| stats values(dpc) as "port (count)" by
dest_ip
| sort +dest_ip
```

dest_ip	port (count)
10.0.0.255	137 (179)
10.100.0.97	514 (4956)
192.168.1.2	53 (11840)
192.168.1.3	53 (9632)
192.168.1.5	123 (2677) 25 (2176) 80 (790)
192.168.1.7	123 (3395) 37710 (1) 39202 (1) 46798 (1) 55338 (1) 58488 (1)
192.168.1.9	111 (2) 123 (2884) 335 (1) 417 (1)
192.168.1.40	123 (1167)
192.168.1.41	123 (1268)
192.168.1.157	41320 (1)
192.168.1.160	123 (1272)
192.168.1.255	137 (2260) 138 (1864)

Visualization

splunk> .conf18

Security Analytics

Lateral Movement and Ransomware

splunk> .conf18

Lateral Movement over Common Ports

Explores connections made between internal hosts using commonly-exploited services

- Use Internal networks on both sides of the flow
- Chooses dest ports from common list of exploited services

Search Syntax

```
sourcetype=netflow
| `is_internal_net(src_ip)`
| `is_internal_net(dest_ip)`
| where dest_port in
(22,23,25,53,80,123,135,443,445,3389,5900,5901)
| stats sum(count) as count by src_ip, dest_ip,
dest_port
| sort +dest_ip,+dest_port,+src_ip
```

20 Per Page ▾ Format

src_ip	dest_ip	dest_port	count
192.168.1.5	192.168.1.1	53	376
192.168.1.7	192.168.1.1	53	1554
192.168.1.9	192.168.1.1	53	13555
192.168.1.37	192.168.1.1	53	4
192.168.1.38	192.168.1.1	53	4
192.168.1.40	192.168.1.1	53	194
192.168.1.41	192.168.1.1	53	8
192.168.1.151	192.168.1.1	53	43
192.168.1.152	192.168.1.1	53	33
192.168.1.153	192.168.1.1	53	48
192.168.1.154	192.168.1.1	53	40
192.168.1.155	192.168.1.1	53	1036
192.168.1.156	192.168.1.1	53	44
192.168.1.157	192.168.1.1	53	43
192.168.1.158	192.168.1.1	53	49
192.168.1.9	192.168.1.1	80	2
192.168.1.9	192.168.1.1	443	58
192.168.1.1	192.168.1.2	53	11841
192.168.1.1	192.168.1.3	53	9633

About Support File a Bug © Splunk Inc. All rights reserved.

Visualization

splunk> .conf18

Potential Data Exfiltration

Looks for outbound data flow over HTTP/S

- Uses the base “session stitching” technique introduced above for HTTP/S
- Looks for server-heavy sessions (more bytes sent than received)
- Summarizes by ASN of recipient and internal client

Search Syntax

```
sourcetype=netflow
|`session_stitch_http`
| `is_internal_net(client_ip)`
| where sc_bytes > 2*cs_bytes
| lookup asn ip AS server_ip
| stats sum(sc_bytes) as totalbytes by
client_ip, autonomous_system
| eval sent_mb=round(totalbytes/1024/1024,1)
| where sent_mb>1
```

client_ip	autonomous_system	sent_mb
192.168.1.101	Akamai Technologies, Inc.	373.4
192.168.1.101	Microsoft Corporation	70.4
192.168.1.12	Highwinds Network Group, Inc.	22.3
192.168.1.151	Canonical Group Limited	36.6
192.168.1.152	Canonical Group Limited	76.1
192.168.1.152	Digital Ocean, Inc.	3.8
192.168.1.153	Amazon.com, Inc.	3.3
192.168.1.153	Canonical Group Limited	75.6
192.168.1.153	Cloudflare Inc	1.3
192.168.1.153	Google LLC	1.1
192.168.1.153	OVH SAS	1.1
192.168.1.154	Canonical Group Limited	71.9
192.168.1.155	Canonical Group Limited	75.3
192.168.1.155	Dropbox, Inc.	29.1
192.168.1.155	ESnet	2435.9
192.168.1.156	Canonical Group Limited	76.2
192.168.1.157	Canonical Group Limited	71.8
192.168.1.158	Canonical Group Limited	73.1
192.168.1.35	Microsoft Corporation	1.9
192.168.1.37	Grande Communications Networks, I	37.7

Visualization

splunk> .conf18

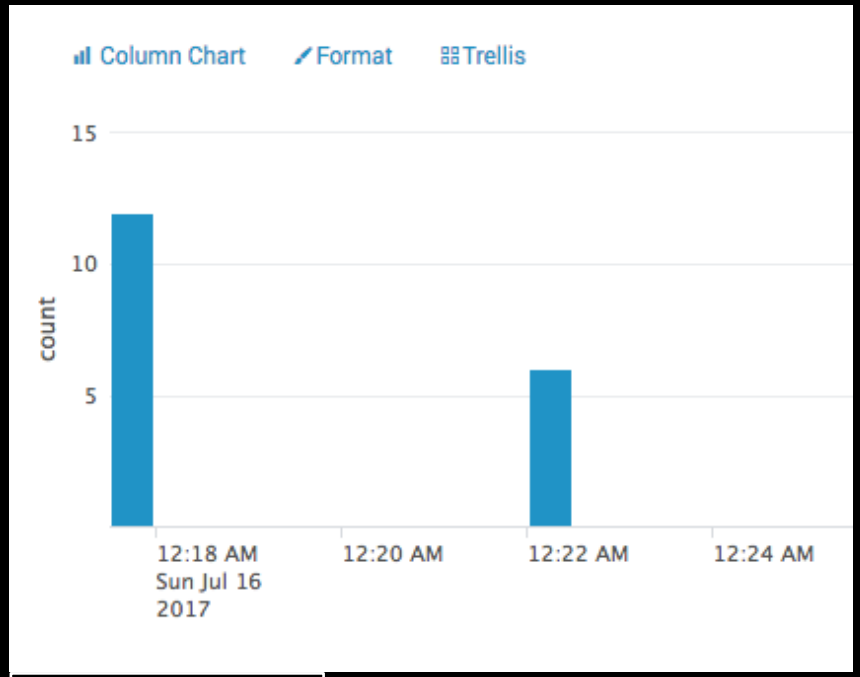
Ransomware Detection

Finds multiple SMB connections between two hosts in short time periods

- Looks over known SMB ports (139, 445) for ransomware attacking file servers
- For multiple connections in a short period of time
- Doesn't work on newer versions of SMB, where persistence is enabled, but v1 and v2 should work

Search Syntax

```
sourcetype=netflow
| where dest_port in (139,445)
| bucket span=1m _time as timebucket
| stats count by src_ip, dest_ip, timebucket
| eval timebucketstring =
strftime(timebucket,"%Y-%m-%dT%H:%M:%S")
| where count >=6
```



Visualization

splunk> .conf18

Wrap Up

Information Density

How do make practical use all this data?

- NetFlow is voluminous
  - Individual records contain very little information
  - However, large volumes are quite valuable
  - But still difficult to make use of
- So how do we deal with it?!
  - Summary Indices
  - Data Model Acceleration
- Licensing?
  - Discounts discounts discounts – ask your RSM!

Key Takeaways

Just skip to this Slide

1. Hi Volume/Lo Value = Information Density
2. Gives you critical visibility for day-to-day operations
3. Volume Pricing incentives (Talk with your RSM!)

Thank You

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