The House Always Wins: Using Splunk Enterprise to Fight Data Exfiltration From Insider Threats

David Doyle | Eric Secules

Bechtel CIRT

Forward-Looking Statements

During the course of this presentation, we may make forward-looking statements regarding future events or plans 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 may differ materially. The forward-looking statements made in the this presentation are being made as of the time and date of its live presentation. If reviewed after its live presentation, it may not contain current or accurate information. We do not assume any obligation to update any forward-looking statements made herein.

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 functionalities described or to include any such feature or functionality in a future release.

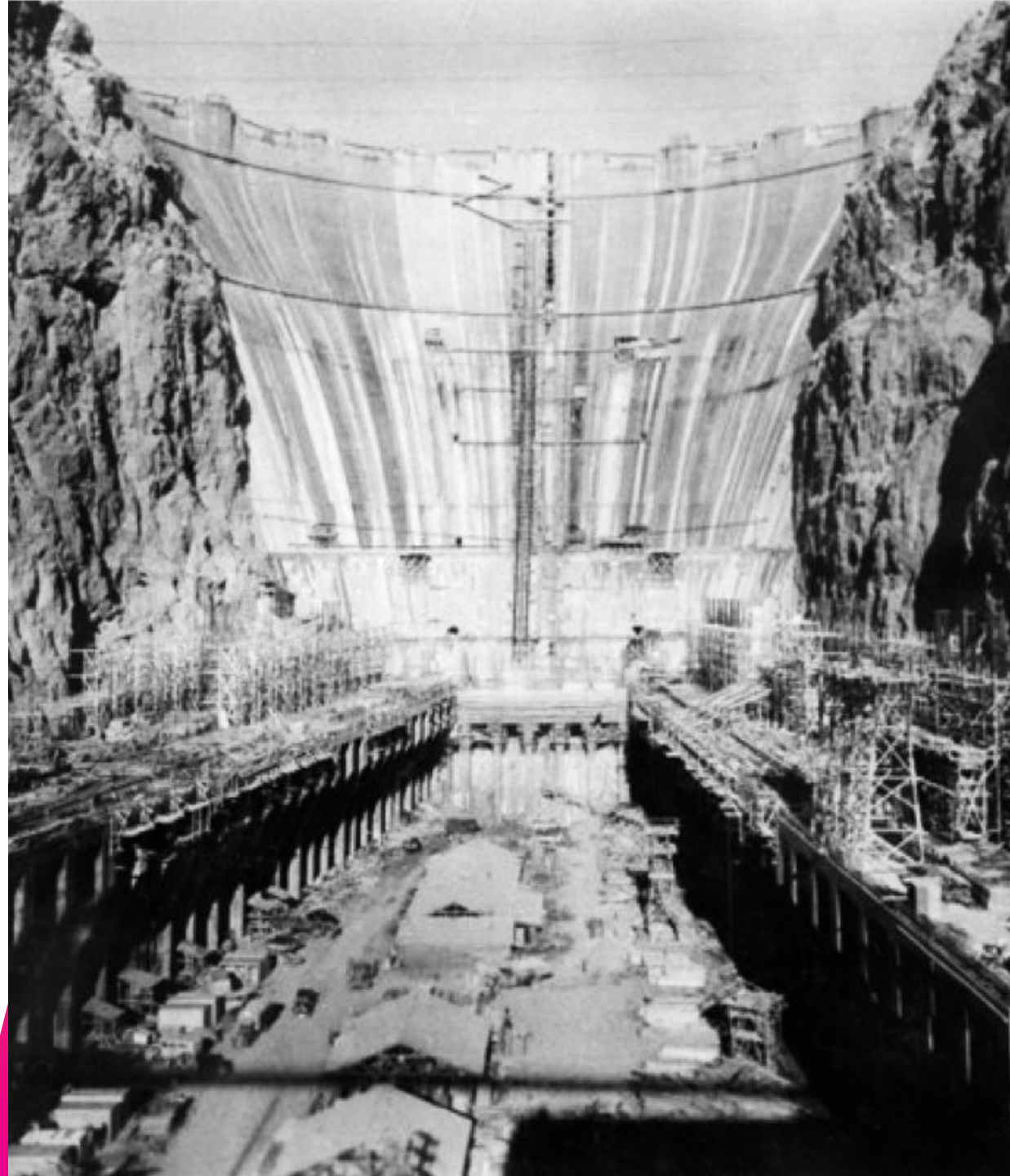
David Doyle

Splunk Puncher | Bechtel
"I want free tickets to .Conf!"

Eric Secules

Forensic Investigator | Bechtel | @nibble4n6
"I want to enhance my insider threat investigations!"

splunk> .conf19



WE BUILD BIG THINGS. EVERYWHERE.

splunk> .conf19

Core Themes

Splunk Enterprise is an effective tool for your Insider Threat toolkit

- Properly implemented, Insider Threat countermeasures can drop your investigative timeframe significantly

Like all tools, Splunk must be calibrated correctly to be effective

- Lot of pregame work involved there, but some has already been done
  - Know your environment, your stakeholders, your knowledge holdersPrevious experience can (and should) shape future actions
- Pulling data manually? Automate it
- Running searches manually? Automate them
- Did something work for your last few investigations? It probably will for your next ones, too
  - Automate it, then look at similar data sources for similar threats, and build some dashboards based on those threat families

Managing Expectations

What it isn't:

A once-size-fits-all solution

- Anyone who says they have that is selling something

Basic "how-to" for Getting Data In

Us saying we know everything about insider threat

What it is:

A post-mortem

- Maybe even an intra-mortem?
- We're still learning

A single set of use cases

- Built for our:
  - Company
  - Industry
  - Environment

Transition

What we used to do, and why we changed it

splunk> .conf19

Reactive Insider Threat Program

Insider Threat "program" was ad hoc and after the fact

- Requests to investigate departed employees sometimes came weeks after employees left
- Difficult to interview someone when they don't work for the company anymore
  - No reason to be cooperative after the fact

Clock was ticking, and in many cases, it had already ticked

- If data leaves the org, no telling what happened with it after the fact

Impetus to Change

We can't always control when we find out that someone is leaving

- Notices aren't mandatory
- Notices aren't always followed
- Involuntary terminations are a thing – people get fired

Policy hurdles

- High-risk users can be locked out from exfil methods
- But this doesn't always happen

But we can reduce the time between notification and reporting

- And if we're lucky, we can identify potential behaviors before they become actual behaviors

Goals

Data-driven investigations

- Build out knowledge objects around known indicators and activity

Reduce timeframes

- Alert to need for investigation
- Begin investigation
- Complete investigation

Reduce start time

- Move from after exfil occurs to before when possible

Gather data for calls to action

- Track more events, track them better
- Use data to push for policy changes

Implementation

Pregame work: inputs, filtering, modeling

Two Basic Options

Vendor-based solution

+ already built

- $$$

- "one size fits all" more like "YMMV"

Go it alone

+ $$$

+ as flexible as we are

- takes time to develop

Going it Alone: What Does That Actually Mean?

Not using a vendor-supplied solution

- No solutions found that satisfied our requirements
  - Lots of false positives
  - Lots of care and feeding
  - Little actual ROI

Lots of institutional knowledge in our organization

- History in the org of rolling things ourselves off Splunk Enterprise
  - We're not crazy, so feeding in from outside tools, feeding out to outside tools

Boiling it Down: What We Actually Use

User definition data

- User id, real name, location, position, etc.
- Further filters define user based on risk potential
  - Employment state: 2-week notice, temporary employee, contractor
  - High-risk locations: projects, government, locations with...a history

Activity data

- "Action" data attributable to a user, their assets, etc.
  - Your classic Splunkable time-series events that you're already pulling in
- Filters here define activity based further based on risk potential
  - This is where most of our "investigative" SPL lives

All data normalized where possible

- This will be important later

But to Get to That Point...

Ask yourself a few questions:

- What log sources have you used in the past in your reactionary investigations?
- What indicators of insider threat and exfiltration are you not looking at?
  - Are you missing data for them, or are you just not looking?
- What do you want your final product to look like?
  - Single-panel reports? Dashboards? Alerts?
  - Who is the audience? How well do THEY understand the data?

Extends to more than just a choice of data to display

Field titles, choice of visualization, alert fatigue, etc.

You Don’t Have All The Data: Engage Others

Data owners:

- Figure out what you have, what you don't have, and who owns the latter
  - Engage and convince, but be prepared to have to get buy-in from higher up the food chain

Might need to argue for a one-way street – not all data owners are investigators, nor should they be

  - Potentially more than just raw data

Do the owning platforms do any useful processing that will save you time/effort?

Splunk owners (assuming that isn't you):

- How to get the data in, where to send it, will it impact licensing, who needs access to it
- When should you be running generating searches, CIM compliant field extractions, non-standard viz TAs

Searching, Visualizing, and Alerting

Or, the part you actually care about

splunk> .conf19

Putting the Pieces Into Place

Activity data

- USB writes
- Emails sent externally
- Large file uploads
- Email forwarding rules
- Cloud shares
- Mailbox archives
- ...and more!

Attribution data

- User identification data
- User state data

Now that you have these, you're going to start generating SPL...finally

Putting the Pieces Into Place: User Data

By far, the most varied area of implementation

- Pulling from lots of environment-specific sources
  - Splunk data
  - HR databases
  - Active Directory
  - [MORE TBD]
- Using lots of tool-specific methods
  - SPL
  - DB Connect
  - Powershell
  - [MORE TBD]
- Your mission: formalize, normalize, and add risk scores

Putting the Pieces Into Place: User Data

SPL example: emails sent externally

```
index=[email_external] userid="*@yourdomain.tld"
| rex field=userid "(?<user>[^@]+)"
| stats count as externalemail by user
| where externalemail > [threshold]
| eval user_risk = [value]
| eval query = "external_email"
| eval run_time = strftime(now(), "%Y/%m/%d %H:%M:%S")
| table user, externalemail, user_risk, query, run_time
| outputlookup externalemail.csv
```

Putting the Pieces Into Place: User Data

Pulling characteristics from AD

- Schedule Powershell searches to run at regular intervals on a box connected to AD (before Splunk runs its scheduled searches for dashboarding/alerting)
  - Have search results picked up by Splunk

Accounts younger than one year:

- get-aduser -filter * -Properties whencreated | where {$_.whencreated -gt $((Get-Date).AddDays(-365))} | Select samaccountname | export-csv C:\output\shorttermusers.csv

Higher risk office/project locations

- "OU=LAS,OU=ABCDE" | foreach {Get-ADUser -filter {enabled -eq $True} -SearchBase "$_,OU=01_Users,dc=abcde" -Server IABCDE | Select SamAccountName} | export-csv C:\output\highriskusers.csv

The Great [REDACTED] Opsec Slide

We can't really talk about every data source

- Neither could you, if you were up here

You're already doing investigations

- Aggregate data from them
- Develop trends
- Build models
- Integrate them!

Putting the Pieces Into Place: User Data

Stapling all your sources together:

```
| inputlookup input1.csv
| inputlookup input2.csv append=t
| search user=$user_token$
| streamstats last(FutureTermDate) as FutureTermDate by user
| eventstats last(city) as location by user
| eventstats last(title) as jobtitle by user
| dedup user query
| eval run_epoch = strptime('run_time', "%Y/%m/%d %H:%M:%S")
| eval earliest_epoch = $it_time_earliest$
| where run_epoch >= earliest_epoch
| eventstats sum(user_risk) AS total_score by user
| fields user, total_score, FutureTermDate, jobtitle, location, query
| mvcombine query
| rename user AS User total_score AS "Total Score" FutureTermDate AS "Termination Date" jobtitle AS
"Job Title" location AS Location query AS "Threat Vector"
| table User "Total Score" "Termination Date" "Job Title" Location "Threat Vector"
| sort 0 -"Total Score"
```

Putting the Pieces Into Place: User Data

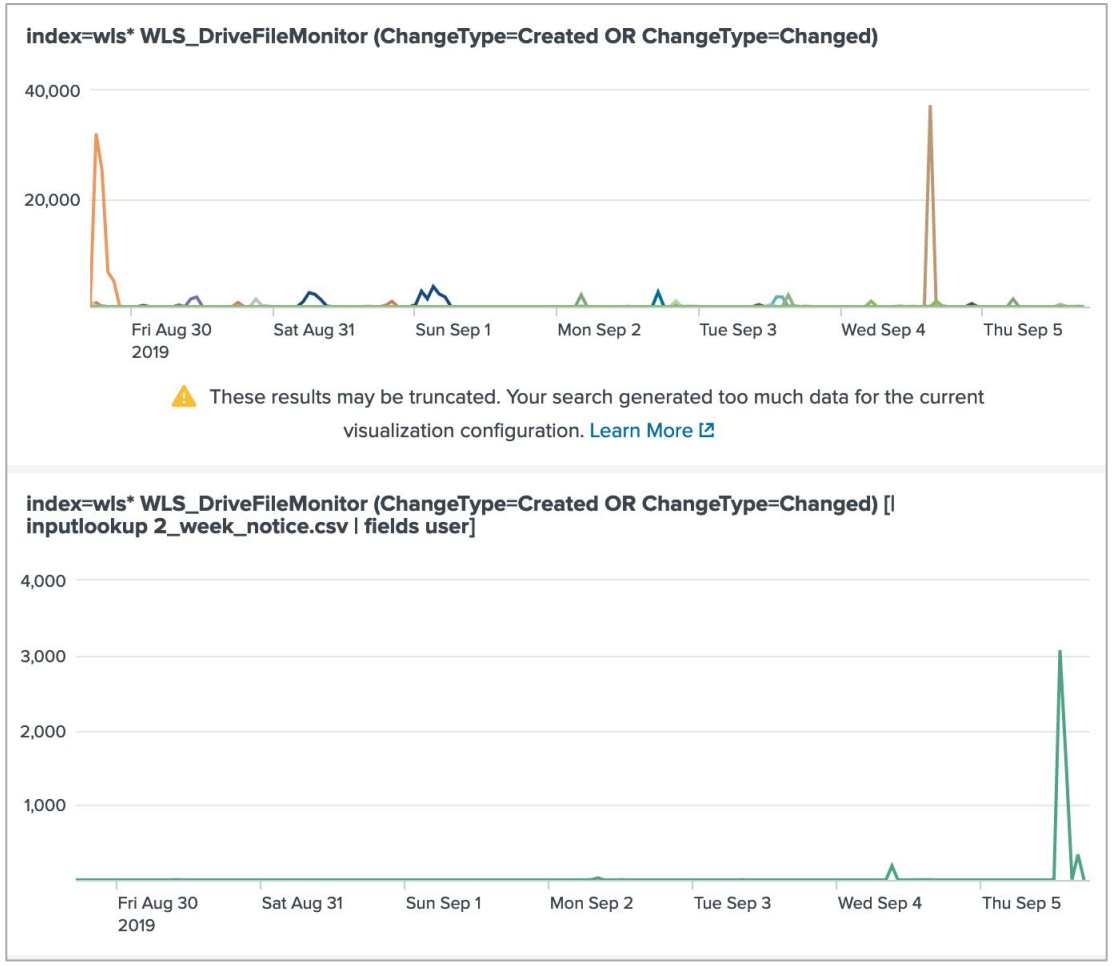
High Risk User

User	Total Score	Termination Date	Job Title	Location	Threat Vector
[redacted]	35	20190901	NP Build Cont Supv Structural	[redacted]	futureterm external_email high_risk_geo large_xfer
[redacted]	20	20190901	Casthouse Pre-Comm Automation Test Eng	[redacted]	futureterm high_risk_geo
[redacted]	20		Productivity & Performance Superintenden	[redacted]	external_email high_risk_geo large_xfer
[redacted]	20		Sr Startup Engineer Program Support	[redacted]	external_email large_xfer

Example Use Cases (Finally)

splunk> .conf19

Use Case: Why We Filter



Use Case: Cloud Traffic Patterns, Upload/Download, Single Source

Example dataset: company-owned cloud tenant (O365)

- Advantages include a more robust border to be monitored
- Disadvantages include visibility into events without ability to change policy regarding sharing

Because you have cross-tenant visibility, you can much more than just uploads

- Sharing events
  - Sharing to non-company accounts
- Uploads, downloads, file syncs
  - File syncs =/= uploads and downloads, but they kinda do from your perspective

Make sure you account for them

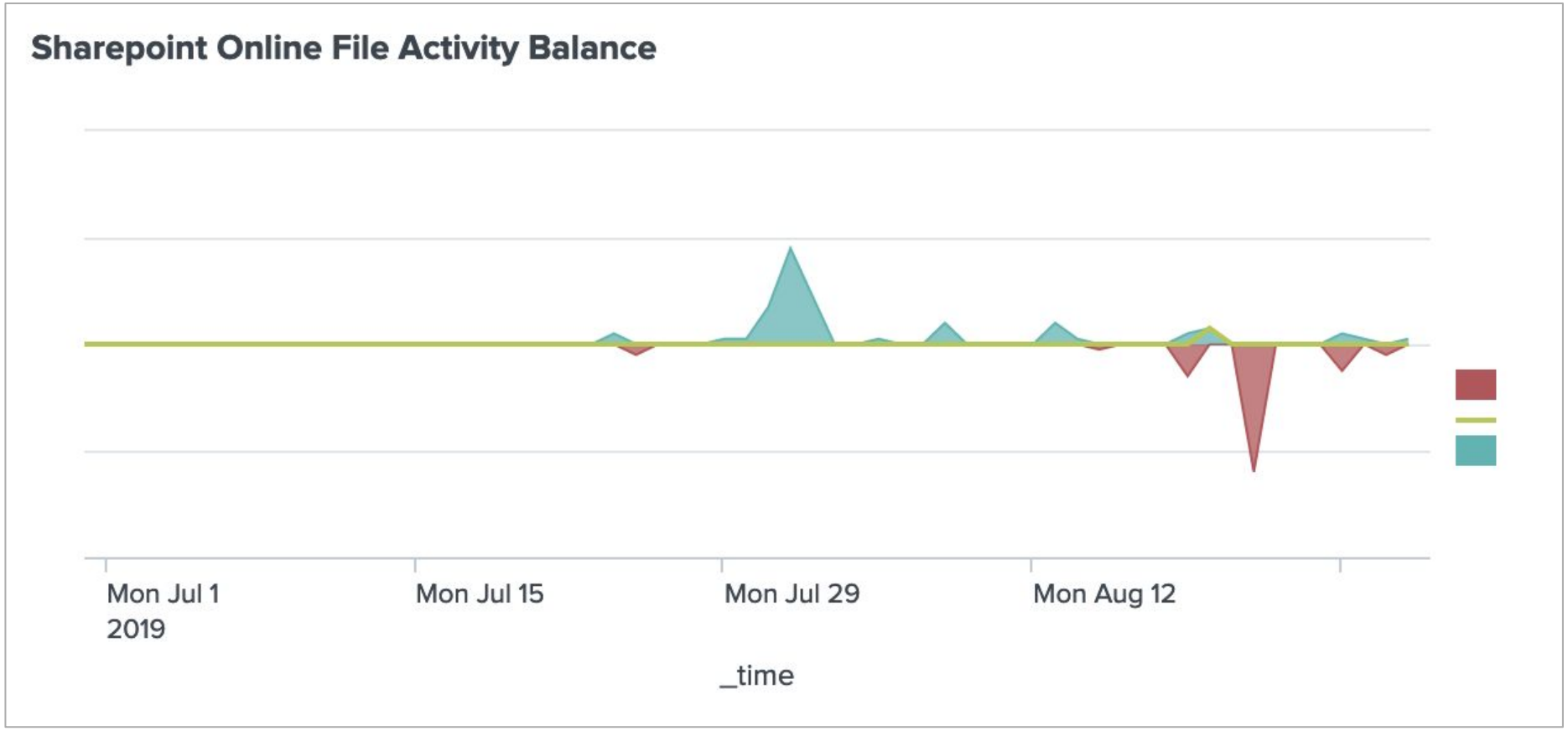
Use Case: Cloud Traffic Patterns, Upload/Download, Single Source

Splunk_ta_o365*, then:

```
index=o365 sourcetype=o365:management:activity UserId=$user_token$ (Operation=*Upload* OR
Operation=*Download*) OR (Operation=SharingInvitation* OR Operation=*SecureLink*
TargetUserOrGroupName!=*@yourdomain.tld)
| rex field=Operation mode=sed "s/^.+Upload.+$/Uploads/g"
| rex field=Operation mode=sed "s/^.+Download.+$/Downloads/g"
| rex field=Operation mode=sed "s/^.+Sharing.+$/Shares/g"
| rex field=Operation mode=sed "s/^.+SecureLin.+$/Shares/g"
| strcat UserId ":" Operation user_ops
| bin span=1m _time
| stats count as ops by _time user_ops
| eval ops=if(match(user_ops,"Upload"),ops,-ops)
| eval ops=if(match(user_ops,"Share"),abs(ops),ops)
| timechart limit=0 sum(ops) by user_ops
```

* How to O365? See https://conf.splunk.com/files/2018/recordings/office-365-in-nearly-sec1097.mp4

Use Case: Cloud Traffic Patterns, Upload/Download, Single Source



Use Case: Cloud Traffic Patterns, Upload/Download, Multiple Sources

Firewall logs can profile traffic

- Profiled traffic can be used to filter by traffic category
- So, use that!

Advantages and disadvantages:

- Multiple cloud traffic sources in one query
- Additional filters available
  - Excessive bandwidth, client-server vs. browser-based, "evasive"
- If you don't own the platform (previous), you don't have the data once it moves off your assets

Use Case: Cloud Traffic Patterns, Upload/Download, Multiple Sources

SplunkforPaloAltoNetworks, then

```
index=[firewall] "app:subcategory"="file-sharing" user=$user_token$
| eval kb_in=bytes_in/1024
| eval kb_out=bytes_out/1024
| eval kb_in=-kb_in
| timechart span=1h sum(kb_in) as "downloads (KB)", sum(kb_out) as "uploads (KB)"
by application useother=0 limit=0
```

Other options for charting include

- by app:excessive_bandwidth
- by app:prone_to_misuse
- by app:evasive

Use Case: Cloud Traffic Patterns, Upload/Download, Multiple Sources

Use Case: Multi-stage Activity

Remember your kill chain

- Exfil by itself can be too late

Look for staging activity in conjunction with exfil

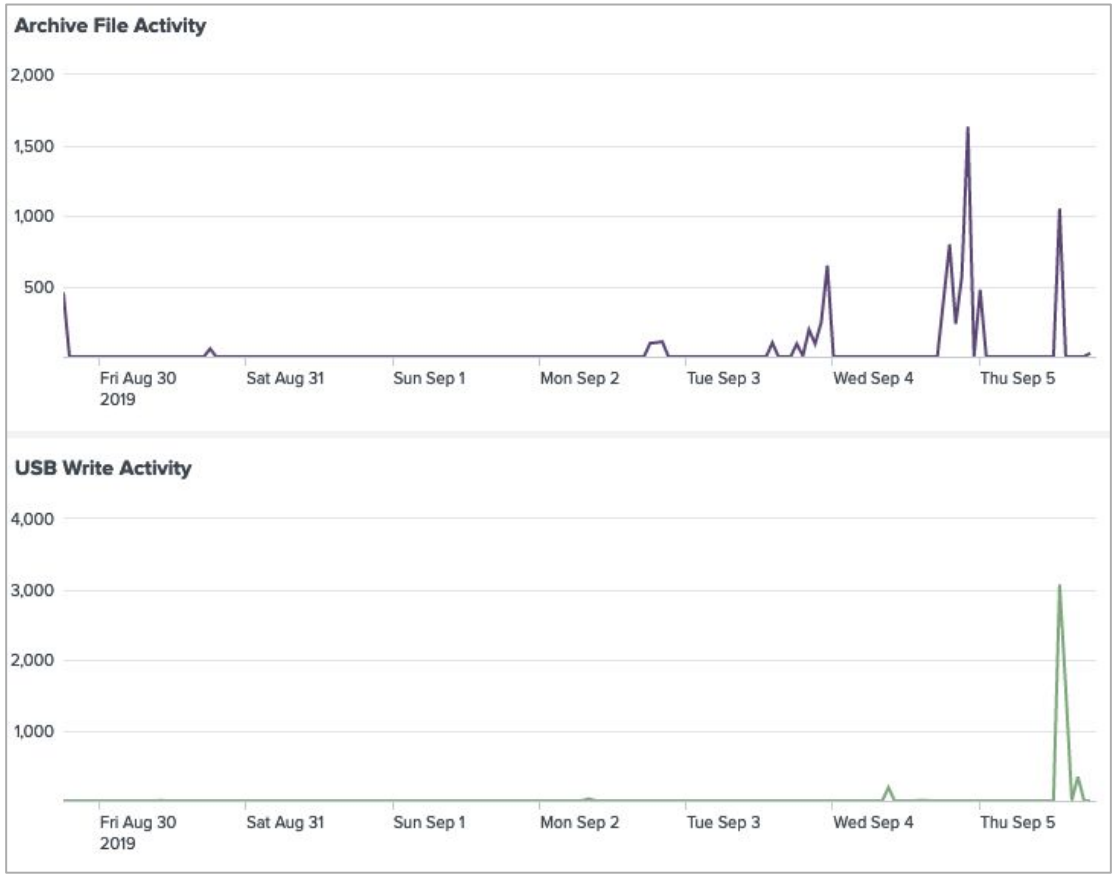
- ZIP-ish archives: .zip, .rar, .7z, .tar/gz/tgz, etc.
- Email archive creation
- Print jobs
- Abnormal physical access
- MITRE ATT&CK technique T1074 has a ton of them

Use Case: Multi-stage Activity

Archive file staging, USB write exfil

- index=[windows] WLS_FileData (BaseFileName="*.7z" OR BaseFileName="*.zip" OR BaseFileName="*.rar") ProcessName!="setup"

- index=wls* WLS_DriveFileMonitor (ChangeType=Created OR ChangeType=Changed)

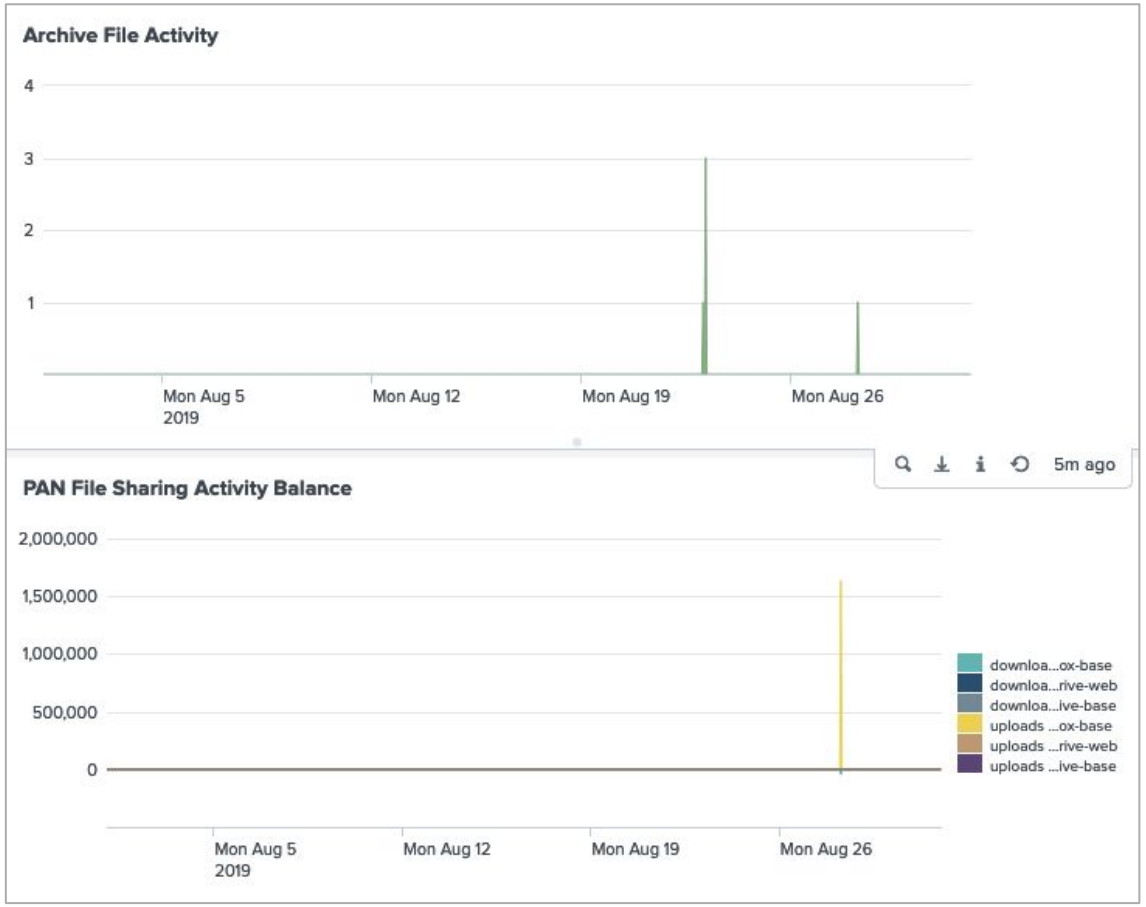


Use Case: Multi-stage Activity

Archive file staging, Dropbox exfil

- index=[windows] WLS_FileData (BaseFileName="*.7z" OR BaseFileName="*.zip" OR BaseFileName="*.rar") ProcessName!="setup"

- index=[firewall] "app:subcategory"="file-sharing"

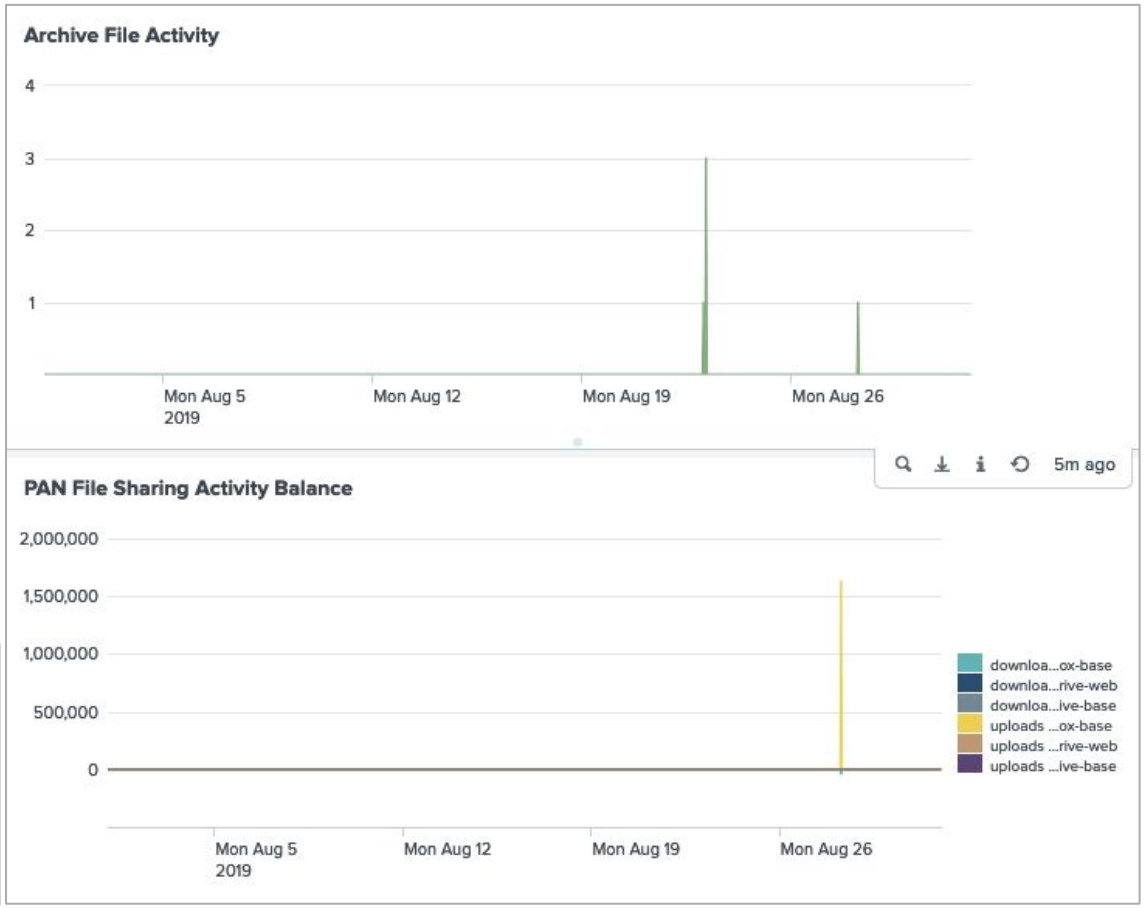


Use Case: Multi-stage Activity

Archive file staging, cloud exfil (Dropbox)

- index=[windows] WLS_FileData (BaseFileName="*.7z" OR BaseFileName="*.zip" OR BaseFileName="*.rar") ProcessName!="setup"

- index=[firewall] "app:subcategory"="file-sharing"



PAN Upload Activity

_time	user	bytes_in	bytes_out	app
2019-08-28 06:46:42	[redacted]	5675003	317099117	dropbox-base
2019-08-28 06:44:24		7690114	381351455	dropbox-base
2019-08-28 06:35:11		3169351	113866628	dropbox-base
2019-08-28 06:26:55		17699523	835750083	dropbox-base

splunk> .conf19

More Staging Events

PST creation

```
index=[windows] Channel=Application ProviderName=Outlook EventId=29
| rex field=Data "Users\\\(<?user>[^\\]+"
```

Print jobs

```
index=[windows] Channel=Microsoft-Windows-
PrintService/Operational (EventID=13 OR EventID=307)
| timechart sum(Pages)
```

Physical access logs

- ...good luck

Industry-specific files

- If your company or industry has crown jewels, watch for them

Use Case: Filtering External Email

Email sent outside of the company

- All outbound email = FP nightmare, even for high-risk individuals

Need another filter; what can we use?

- Previous investigations
- Industry competitors
- Poachers

Turn these into a lookup

- Use lookup as a filter to provide insight

Use Case: Filtering External Email

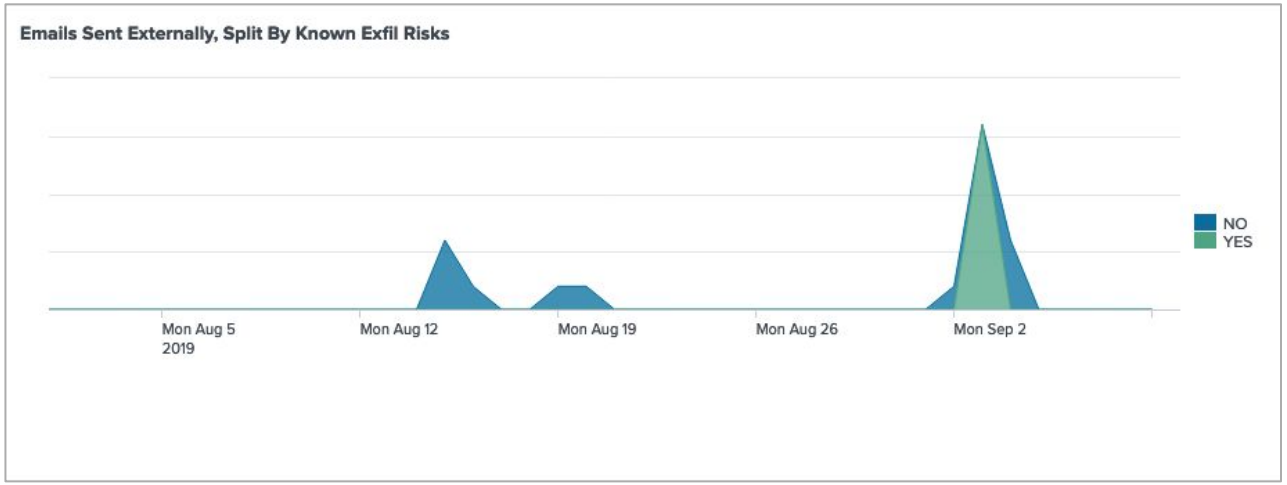
Intel lookup table format:

to_domain	domain_match	domain_category
construxors.biz	yes	competitor
buildabear.ru	yes	competitor
malwarez.gov	yes	malware
wehireurpeeps.com	no	poacher
splunk.com	no	vendor

Useful for other purposes, too....

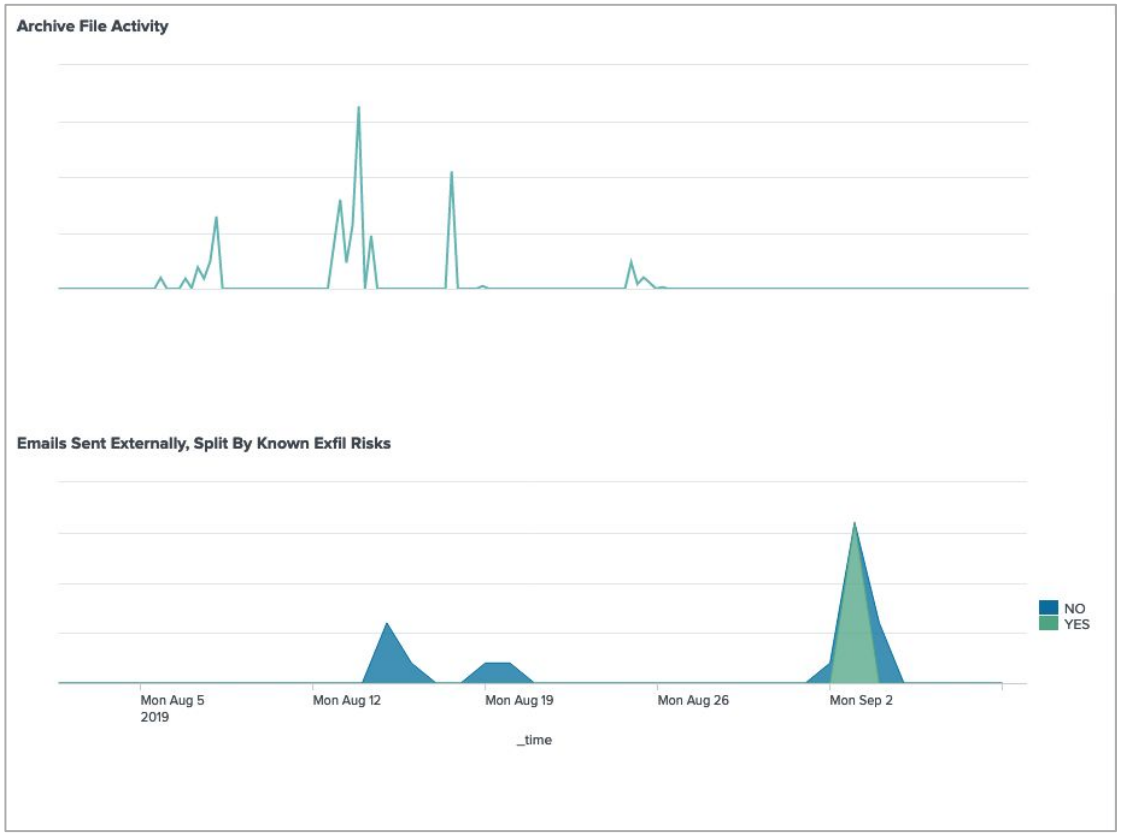
Use Case: Filtering External Email

```
index=ironport
[ search earliest=$timeframe.earliest$
index=ironport
mailfrom=$user_token$@[yourdomain.tld]
    | fields mid
    | format]
| rex field=mailto "@(?<to_domain>.+)"
| transaction mid maxspan=2m
| lookup intel_domains.csv to_domain
OUTPUT domain_match
| fillnull domain_match value="NO"
| search mailfrom=$user_token$@yourdomain.tld
| eval mailfrom=lower(mailfrom)
| timechart span=24h count
```



Use Case: Filtering External Email

```
index=ironport
[ search earliest=$timeframe.earliest$
index=ironport
mailfrom=$user_token$@[yourdomain.tld]
    | fields mid
    | format]
| rex field=mailto "@(?<to_domain>.+)"
| transaction mid maxspan=2m
| lookup intel_domains.csv to_domain
OUTPUT domain_match
| fillnull domain_match value="NO"
| search mailfrom=$user_token$@yourdomain.tld
| eval mailfrom=lower(mailfrom)
| timechart span=24h count
```



Email Bonus Round!

```
index=ironport
[ search earliest=$field1.earliest$ index=ironport mailfrom=$user_token$@yourdomain.tld
| fields mid
| format]
| rex field=mailto "@(?<to_domain>.+)"
| t
| s
| t
```

2019-08-06 09:01:08	naughtyuser@weseeyou.us	user1@knowndomain1.tld	knowndomain1.tld
		user2@knowndomain1.tld	UNKNOWNDOMAIN2.TLD
		user3@knowndomain1.tld	UNKNOWNDOMAIN1.TLD
		USER1@UNKNOWNDOMAIN1.TLD	
		USER2@UNKNOWNDOMAIN1.TLD	
		USER1@UNKNOWNDOMAIN2.TLD	

Putting it All Together: Dashboarding

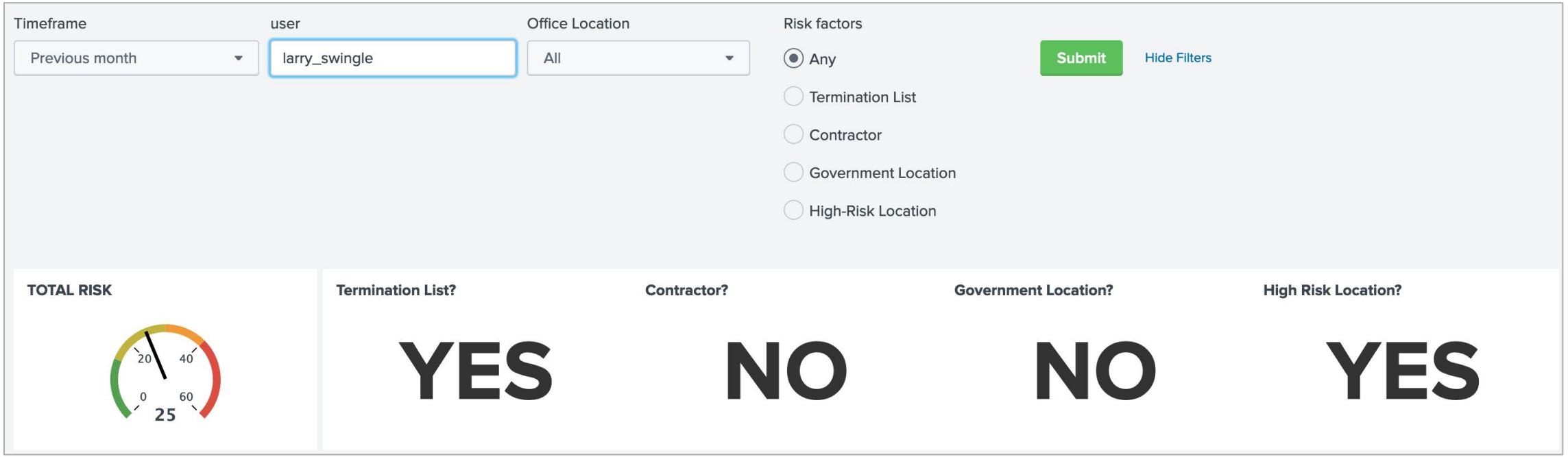
Did you normalize your data?

- If not, good luck with your inputs
- If so, you can split by more than just user
  - Functional Business Units/Departments
  - Offices

Summarizing is easy

- Inputlookups with $user_token$, append, stats sum(user_risk) AS "Total Score" -> radial gauge
- Single inputlookup with $user_token$, eval matched=if(match(user, "^.+"), "YES", "NO") -> single-value YES/NO panel for each threat vector

Dashboarding: Inputs



David's "Keep Your Splunk Admin Happy" Corner

By all means:

Avoid:

Split your dashboards up logically

Run your feeder searches at off-peak hours

Notify Splunk admins in advance of new data

Have answers to questions like "who should have access to this data?"

Dropping 30 panels on one dashboard

Generating all your data at noon on Monday

Asking "where should this go" after the "unknown data incoming" alerts already start

"RBAC? What's that?"

Next Steps

Evolve everything

- Searches, reports, dashboards, alerts
- New data sources

Get even more proactive

- This is now a great dataset to work with
  - Track exfil attempt vectors

(You're already doing that, right?)

  - When common threads emerge, push for preventive measures

Lock out exfil vectors for high risk groups

.conf19

splunk>

Thank You !

Go to the .conf19 mobile app to

RATE THIS SESSION