Add-on Best Practice Check Tool

Brian Wooden

Senior Engineering Manager, Splunk

Jack Coates

Director of Product Management, Splunk

splunk>

Disclaimer

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 the 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.

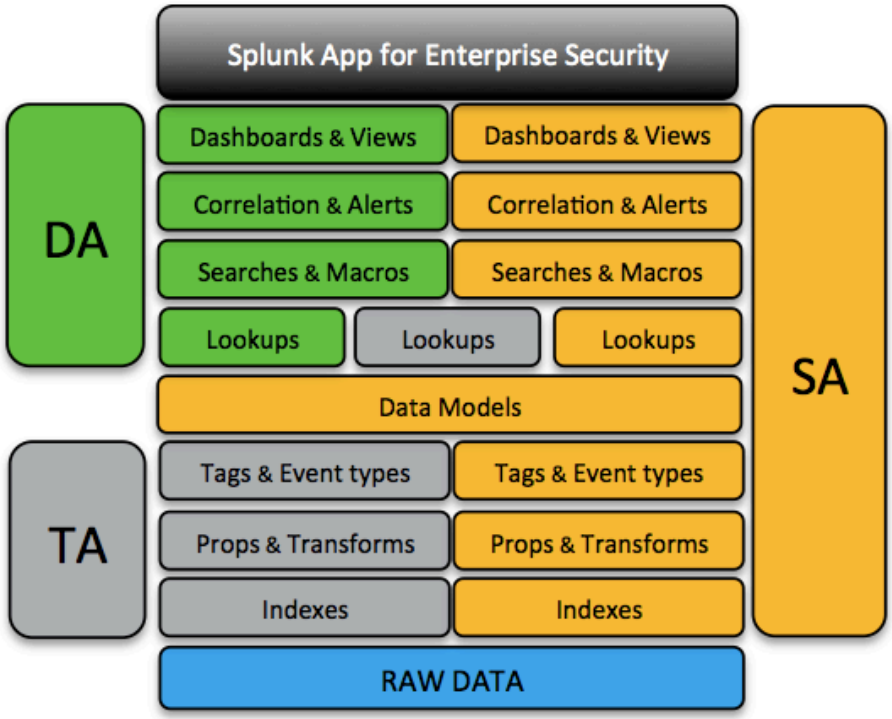
Agenda

- What's a Technology Add-on?
- Naming Conventions
- Getting Data In
- Field Extraction Techniques
- Logging for Troubleshooting
- Data for Sample Events

What's a Technology Add-on?

http://docs.splunk.com/Documentation/ES/latest/Install/ESArchitecture

- It's a driver for non-Splunk technology
  - **Maybe** gets data in, **maybe** sends commands out
- It's a knowledge layer
  - **Definitely** makes sense of the data
- Yes, they're all just app packages
  - Standards that aren't code-enforced are really just guidelines
- Some apps are bigger than others
  - (Some apps' mothers are bigger than other apps' mothers)



Package Names

- The package name is the **folder name** and **app.conf id**.
  - It's hard to change, so get it right before you publish.
  - `Splunk_TA_$vendor-$product` is the format for Splunk-supported.
  - `TA-$vendor-$product` is the format to use if you aren't in Splunk R&D.
- If you use those, you'll be automatically inherited into ES.
- Use the shortest sensible name
- Use lower case
- Try to avoid version numbers
  - Follow this: http://docs.splunk.com/Documentation/Splunkbase/latest/Splunkbase/Namingguidelines

GOOD	LESS GOOD, BUT STILL OKAY	NOT SO GOOD
Splunk_TA_mcafee	TA-Symantec-DeepSight	TA-Damballa-Failsafe-520-v101
Splunk_TA_cisco-asa	Splunk_TA_flowfix	TA_opendns

Source Type Names

- The sourcetype names in props.conf are how knowledge works
  - It's hard to change later, so get it right before you publish
    - Account for backward compatibility with a sourcetype rename
      - If you don't, your Add-on will have to co-exist with the old one as long as data is stored
  - Use `vendor:product:feature:format`
    - Why not underscores? Why not ampersands? Consistency is the goal
  - Use the shortest sensible name
  - Use the lowest sensible number of sourcetypes
  - Break the rules if you have to in order to get data in efficiently

GOOD	LESS GOOD, BUT STILL OKAY	NOT SO GOOD
cisco:esa:textmail	cisco_wsa	syslog
f5:bigip:asm:syslog	Perfmon:sqlserver:locks	microsoft:lync:general

Event Type Names

- The event type names in eventtypes.conf are how CIM tagging works
  - It's hard to change later, so get it right before you publish
    - Special characters (including colons) make startup config testing unhappy
  - Use `vendor_product_eventtype`
    - Consistency is the goal
    - This format is clearly distinguished from sourcetypes
  - Use the shortest sensible name
  - Use the lowest sensible number of event types

GOOD	LESS GOOD, BUT STILL OKAY	NOT SO GOOD
nagios_service_flapping_alert	estreamer_v4	update
snow_change_request	hyperv_host	vnx:block:SystemCpu

Lookup Names

- Lookups are a useful utility and a source of confusion and errors
- Clear naming convention makes them easier to troubleshoot
  - The filename should begin with a short identifier of the add-on, be descriptive, and be plural. `snort_severities.csv`
  - The transforms entry should be the singular form of the filename with `_lookup` added. `[snort_severity_lookup]`
  - The props entry should be descriptive and reference the add-on. `LOOKUP-severity_for_snort`
- Try for shorter names when possible, for btool's sake

Inputs

- Can you control the data that is generated?
  - Review Splunk's Logging Best Practices
  - Time and date entries should always use ISO 8601
- Optimize timestamp recognition as much as possible
- Always disable default inputs
  - Principle of Least Surprise: Enterprise admins want to turn it on themselves
  - Add-ons should get deployed all over, but they shouldn't run everywhere
  - Document how to turn them on where input is desired

disabled = 1

Dependencies

- Need another Add-on for your input? Don't ship an inputs.conf file
  - Common dependencies: DB Connect, PowerShell, Windows, Unix
  - Use `inputs.conf.template` to protect from startup errors
  - Why not a private `.spec` file? Because yours will get out of sync
- Prefer Modular Inputs to Scripted Inputs
  - Why write your own control logic?
- XML Payload? Use a CDATA wrapper
  - Otherwise special characters will break the input
- Asking Splunk REST endpoints for data? Use count=0
  - Otherwise, it will just return a few data points instead of all of them

Indexes Rules

- Do not ship an `indexes.conf`
  - The customer needs to be in control of storage location, data retention, role based access security, and deployment techniques
- Indexes should not be defined without clear understanding of the target Splunk deployment's
  - Security goals & settings (which roles have access to which indexes, &c)
  - Retention goals & settings (restrictions due to disk size, legal, compliance, &c)
  - Storage location, sizes, and performance (local disks, NAS, SAN, allocated free space, IOPS, &c)
  - Index configuration management (deployment server, clustering, puppet, chef, &c)
- Use a macro in your app so that customers can edit

Field Extractions

- The more complex a field extraction is, the worse it will perform
- Calling the regular expression engine repeatedly for simple tasks is better than calling it once for a giant task
  - Readability of `src_port=(\d+)` is better than

```
^\w{3}\s+\d{2}\s+\d{2}:\d{2}:\d{2}\s+\S+\s+(?:\w{3}\s
+\d{2}\s+\d{2}:\d{2}:\d{2}\s+)?([^\t]+)\t([^\t]+)
\t([^\t]+)\t([^\t]+)\t\"?\d+:\s+([^\t"]+)\"?\t([^\t]
+)\t\"?([^\t"]+)\"?\t([^\t]+)\t([^\t]+)\t([^\t]+)
\t([^\t]+)\t([^\t]+)\t([^\t]+)\t([^\t]+)\t\"?([^\t"]
+)\"?\t([^\t]+)\t([^\t]+)
```

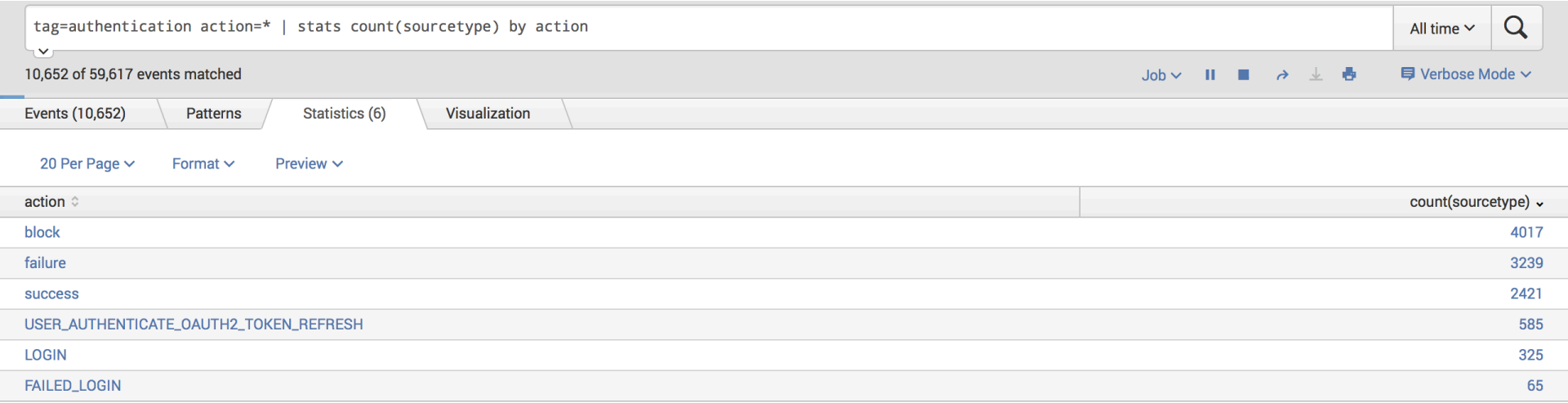
Field Extraction Rules

- Fields used in prebuilt searches (eventtype, savedsearch, view or macro) MUST use static key value extractions instead of variable keys.
- Use the fastest field extraction mechanism that works for the data.
- Do not use variable keys.
  - DM Acceleration and searches are slowed by an order of magnitude
  - You can't CIM map unknown field names anyway

```
REGEX=([^\s=]+)=+(\S+)
FORMAT=$1::$2
```

Over-riding Fields

- Vendors like the same field names that we do. It is to cry.
- Use Field Aliases to move conflicting fields to `vendor_$fieldname`
  - `FIELDALIAS-vendor_action_for_mcafee_epo=action AS vendor_action`
- Use Lookups to resolve conflicts
  - `LOOKUP-action_for_mcafee_epo=epo_action_lookup vendor_action OUTPUT action`



Logging

- Users are taught to expect standard logging behavior from Splunk Add-ons
  - http://docs.splunk.com/Documentation/AddOns/latest/Overview/Troubleshootadd-ons
  - Write log files to locations that Splunk will index, using Splunk best practices for timestamp and format
  - Set at least one sourcetype so that you're not adding to "the learned problem"
  - Write your logs with levels, even if you don't let the user select levels yet.
    - The standard levels are `INFO`, `WARN`, `ERROR`, and `DEBUG`
- Log your intent and the outcome in the log
  - You want to authenticate to a service? DEBUG log that you are attempting it.
  - If it fails, ERROR log that it failed.
  - If it succeeds, INFO log that you're authenticated.

Sample Data and Customer Data

- Don't send us non-anonymized customer data or include in the Add-on
  - This isn't support, it's product development – different rules
- Customer data is normal data from working systems. That's okay for operations, but insufficient for security.
  - Fine for INFERENCE, inherently insufficient for RECOGNITION
    - INFERENCE: "Frank never has a second cup at home!"
    - RECOGNITION: "Frank bought a plane ticket to Colombia for better coffee."

Questions?

THANK YOU

splunk>