Multi-tenancy Management

Andre Tucker
Solutions Architect - ReliaQuest
atucker@reliaquest.com

About Me

- ReliaQuest Solutions Architect

- IT Experience:
  - System Administrator
  - Web Development / Database administration
  - SIEM Content Development
  - Software Development

- 3 year Splunk developer / enthusiast / evangelist

What is Multi-tenancy

Multi-tenancy - an architecture in which a single instance of a software application serves multiple customers.

- *Child Companies*
- *Business Units*
- *Any Sub-Entity*

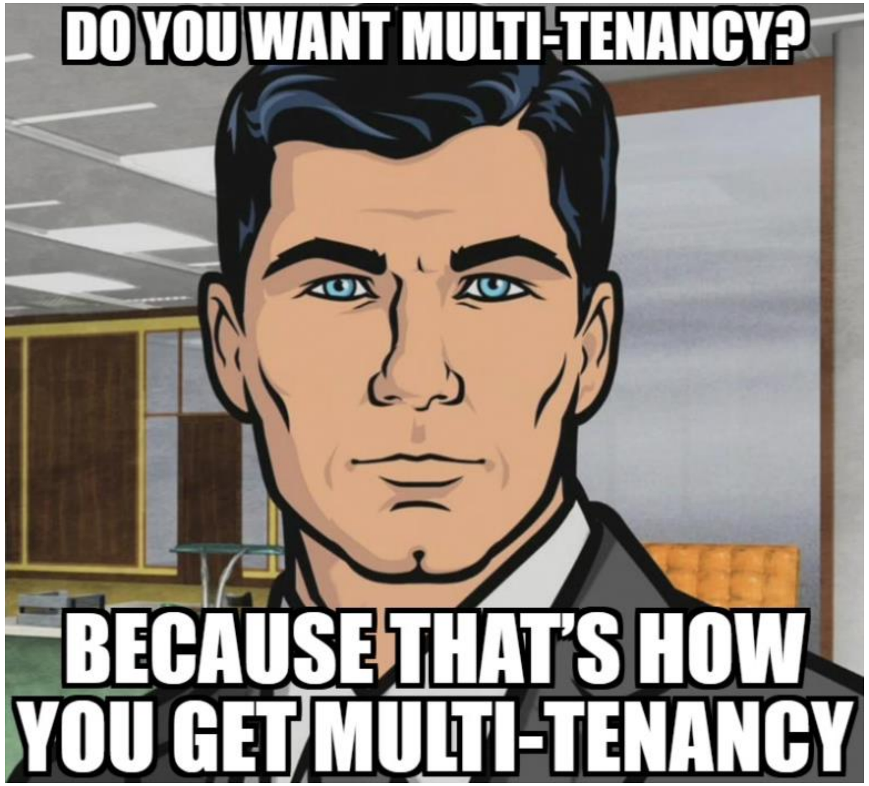
Multi-tenancy Goals

Well executed multi-tenancy should:

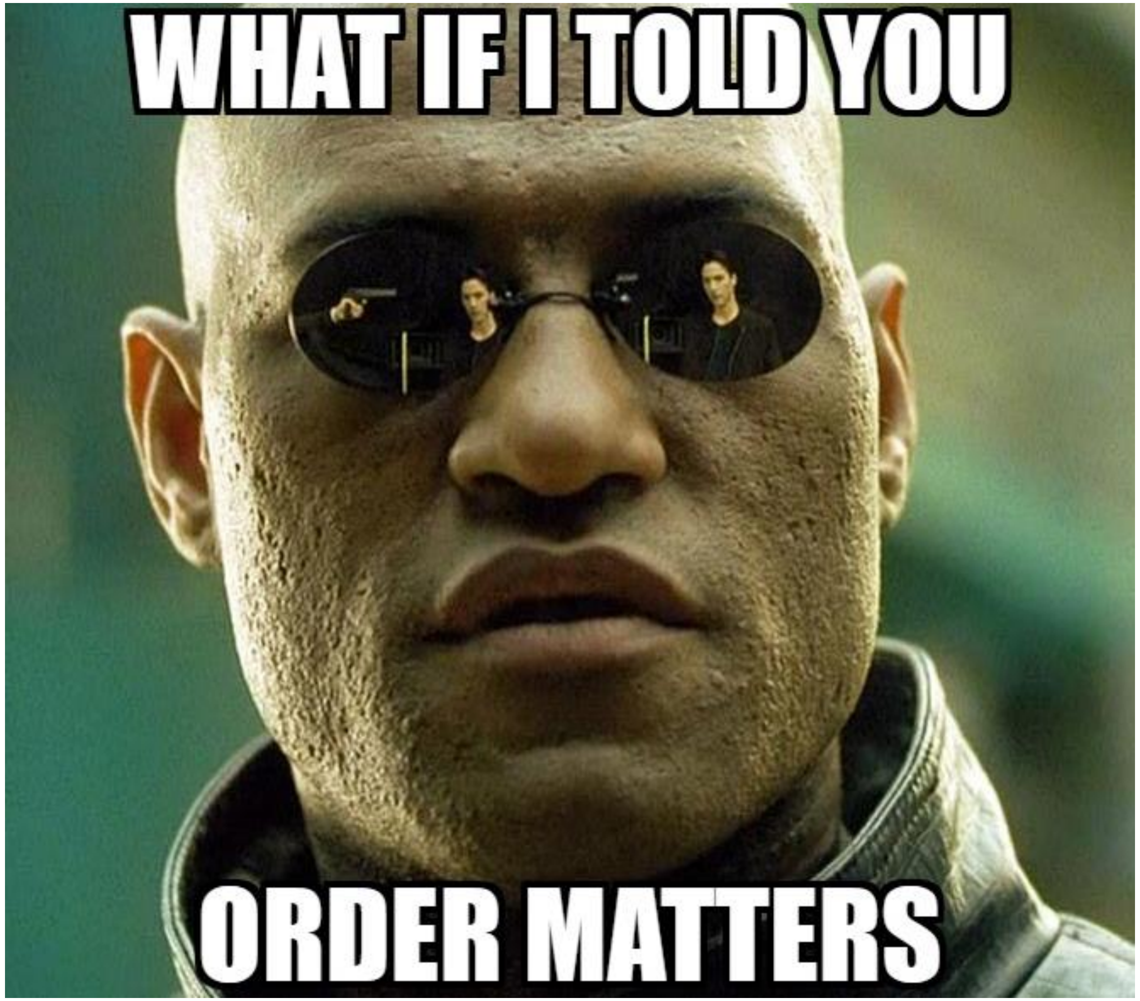
- *Provide complete autonomy for each unit/entity*
    - Data Visibility
    - Content (reports, dashboards, alerts)
- *Minimize content duplication*
    - Granular changes per entity
- *Allow for overall program visibility and conformation*
    - Reporting on groupings of entities.
    - All entities have the conform to enterprise program

The Approach

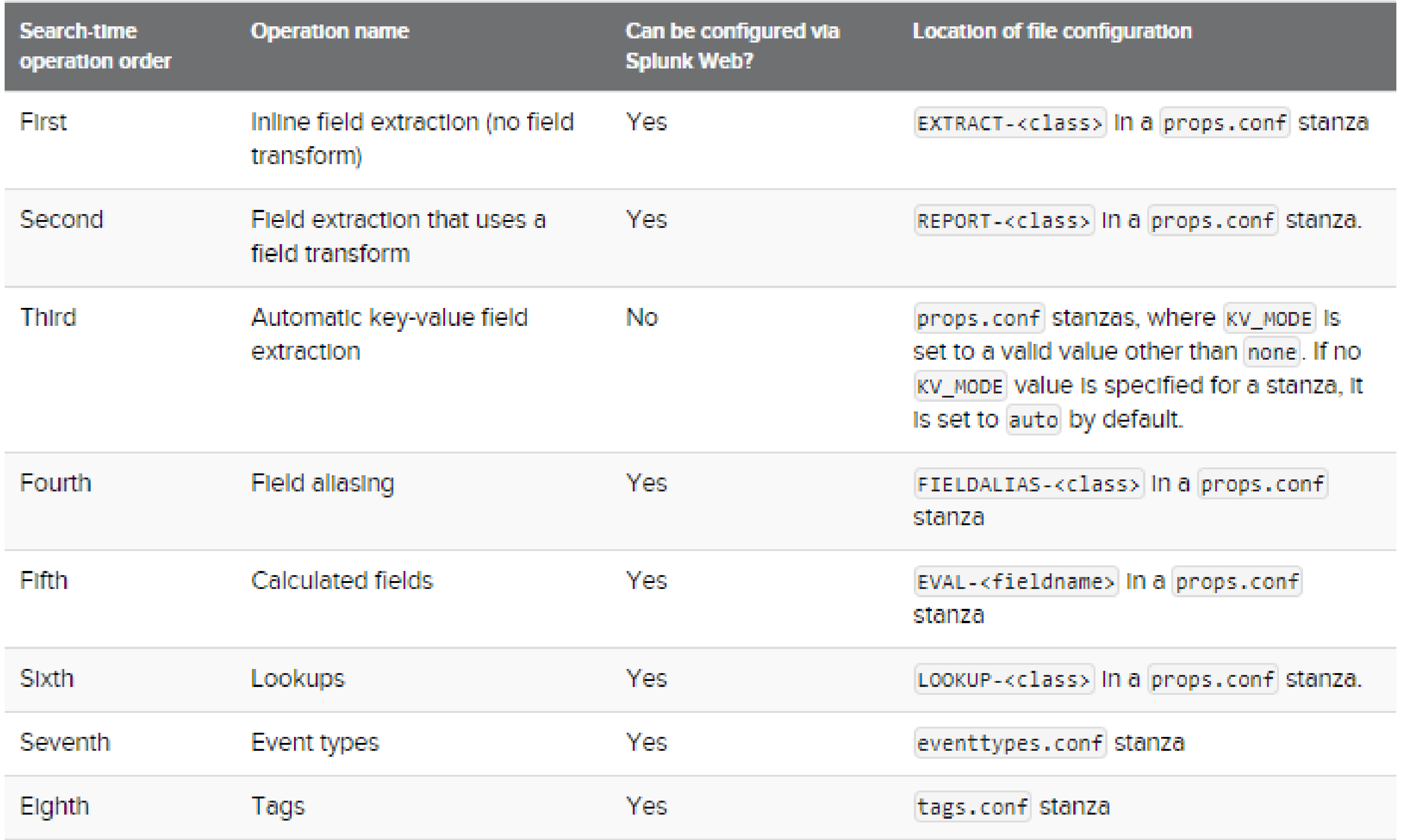
- Data Indexing
- Data Classification
- Data Manipulation


DO YOU WANT MULTI-TENANCY?
BECAUSE THAT'S HOW YOU GET MULTI-TENANCY

The Approach



Order of Operations



Search-time operation order	Operation name	Can be configured via Splunk Web?	Location of file configuration
First	Inline field extraction (no field transform)	Yes	`EXTRACT-<class>` in a `props.conf` stanza
Second	Field extraction that uses a field transform	Yes	`REPORT-<class>` in a `props.conf` stanza.
Third	Automatic key-value field extraction	No	`props.conf` stanzas, where `KV_MODE` is set to a valid value other than `none`. If no `KV_MODE` value is specified for a stanza, it is set to `auto` by default.
Fourth	Field aliasing	Yes	`FIELDALIAS-<class>` in a `props.conf` stanza
Fifth	Calculated fields	Yes	`EVAL-<fieldname>` in a `props.conf` stanza
Sixth	Lookups	Yes	`LOOKUP-<class>` in a `props.conf` stanza.
Seventh	Event types	Yes	`eventtypes.conf` stanza
Eighth	Tags	Yes	`tags.conf` stanza

http://docs.splunk.com/Documentation/Splunk/7.0.3/Knowledge/Searchtimeoperationssequence

splunk> .conf18

Data Indexing

Naming Conventions

- Prefix indices based on entity
- Follow prefix by categorical name

<entity>_<category> = pod1_firewall

Routing

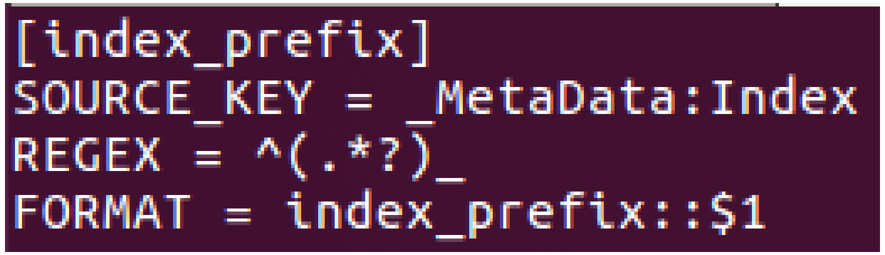
- Route based on host or source

Permissions

- Align indexes with necessary roles

Data Classification

Transforms – First get the entity prefix from the index name

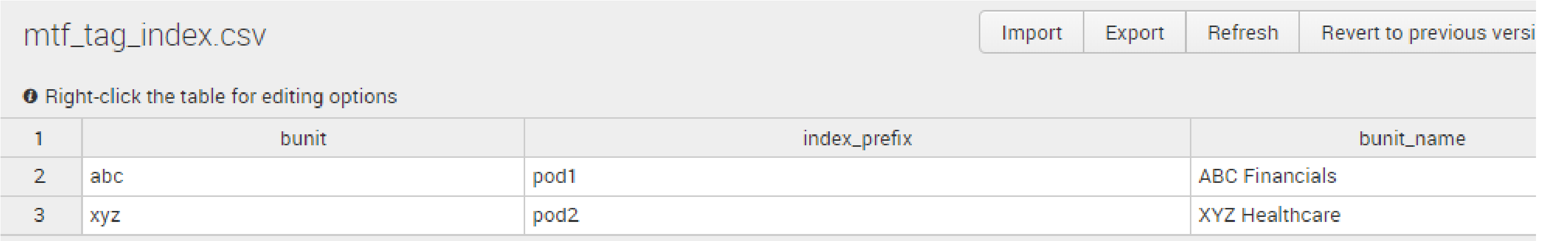


```
[index_prefix]
SOURCE_KEY = _MetaData:Index
REGEX = ^(.*?)_
FORMAT = index_prefix::$1
```

pod1_firewall

Lookups – Enrich Splunk data by adding field-value combinations found in the lookup.

- Match index patterns to entity names



mtf_tag_index.csv

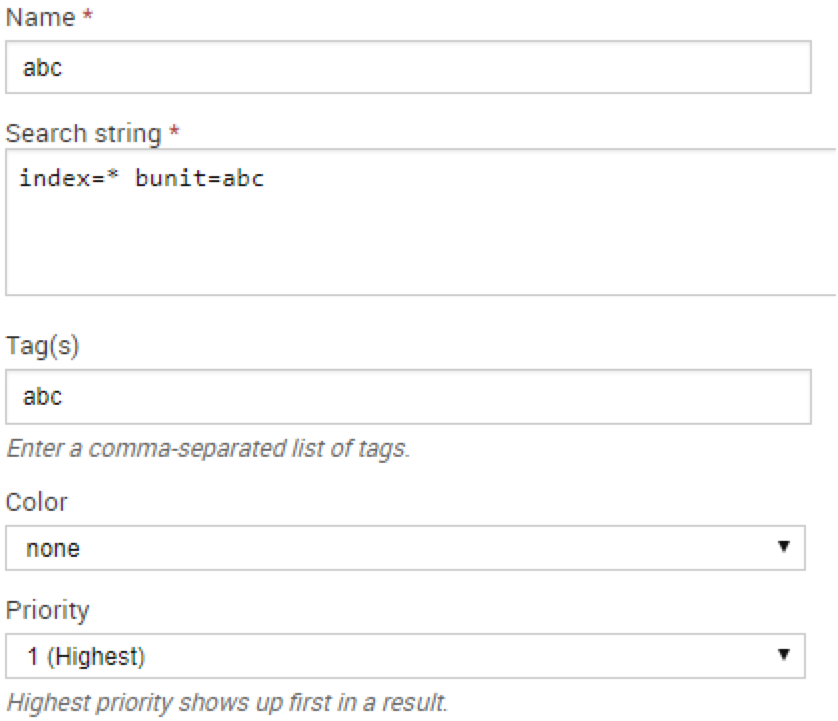
Right-click the table for editing options

1	bunit	index_prefix	bunit_name
2	abc	pod1	ABC Financials
3	xyz	pod2	XYZ Healthcare

Data Classification

Eventtypes – Next we search for all events that should belong to the entity.

Index=* bunit=abc



Name *

abc

Search string *

```
index=* bunit=abc
```

Tag(s)

abc

Enter a comma-separated list of tags.

Color

none ▼

Priority

1 (Highest) ▼

Highest priority shows up first in a result.

Tags – And group them together.

Data Manipulation

- Control *production / development* entities using lookups, tags, and incident review states
- Craft *emails* based on entity using eval
- Control *logic* using a mixture of lookups and macros.
  - (Wanna go crazy? Try the “map” command)
- Change *dashboard* views / panels using tokens

Thank You

atucker@reliaquest.com