Splunk Fundamentals

Outline

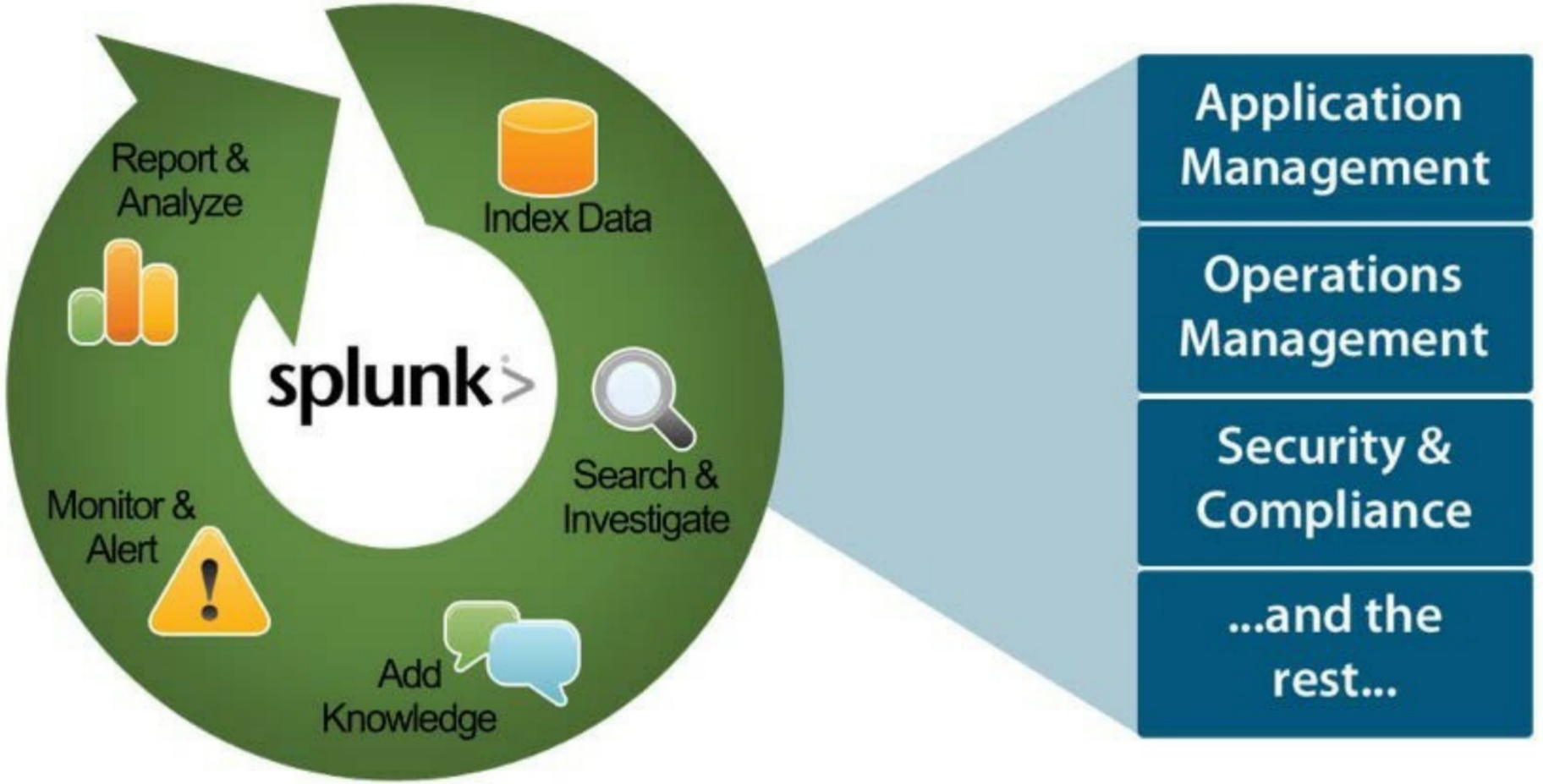
Module 1: Introducing Splunk
Module 2: Splunk Components
Module 3: Installing Splunk
Module 4: Getting Data In
Module 5: Basic Search
Module 6: Using Fields
Module 7: Best Practices
Module 8: Splunk's Search Language
Module 9: Transforming Commands
Module 10: Creating Reports and Dashboards
Module 11: Pivot and Datasets
Module 12: Creating and Using Lookups
Module 13: Creating Scheduled Reports and Alerts

Introducing Splunk

Understanding Splunk

- What Is Splunk?
- What Data?
- How Does Splunk Work?
- How Is Splunk Deployed?
- What are Splunk Apps?
- What are Splunk Enhanced Solutions?

What Is Splunk?



Aggregate, analyze, and get answers from your machine data

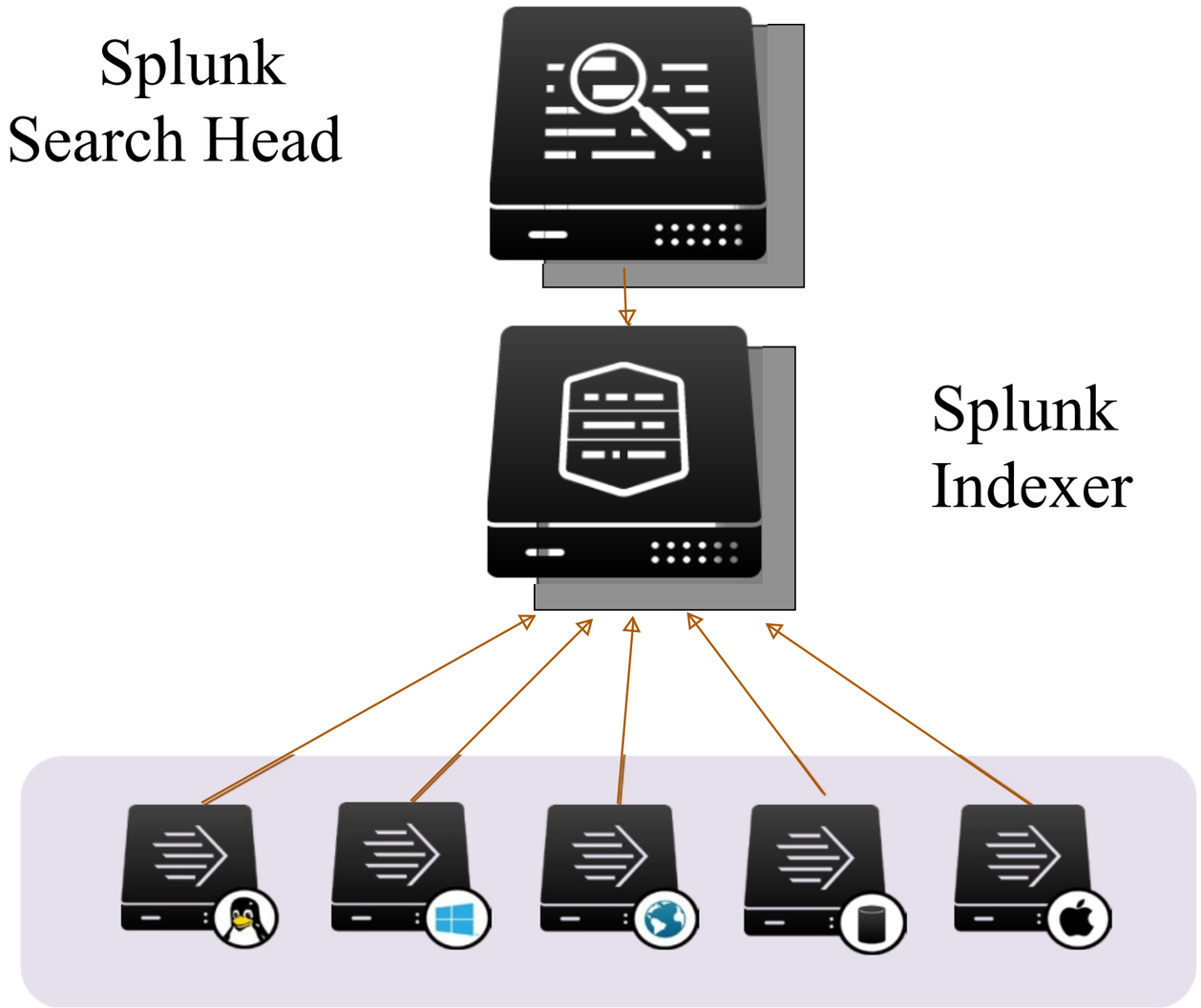
What Data?

Index **ANY** data from **ANY** source

- Computers
- Network devices
- Virtual machines
- Internet devices
- Communication devices
- Sensors
- Databases

- Logs
- Configurations
- Messages
- Call detail records
- Clickstream
- Alerts
- Metrics
- Scripts
- Changes
- Tickets

How Does Splunk Work?



How Is Splunk Deployed?

- Splunk Enterprise

  Splunk components installed and administered on-premises

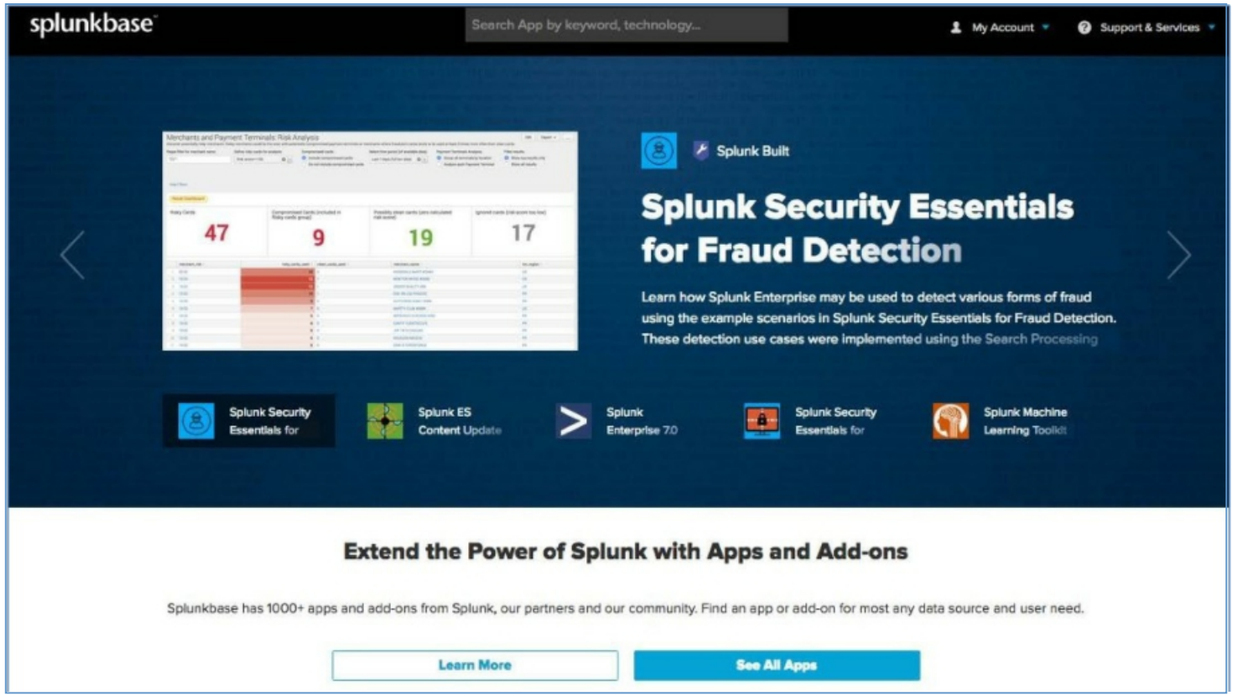
- Splunk Cloud
  - Splunk Enterprise as a scalable service
  - No infrastructure required

- Splunk Light

  Solution for small IT environments

What are Splunk Apps?

- Designed to address a wide variety of use cases and to extend the power of Splunk
- Collections of files containing data inputs, UI elements, and/or knowledge objects
- Allows multiple workspaces for different use cases/user roles to co-exist on a single Splunk instance
- 1000+ ready-made apps available on Splunkbase (splunkbase.com) or admins can build their own



What are Splunk Enhanced Solutions?

- Splunk IT Service Intelligence (ITSI)
  - Next generation monitoring and analytics solution for IT Ops
  - Uses machine learning and event analytics to simplify operations and prioritize problem resolution
- Splunk Enterprise Security (ES)
  - Comprehensive Security Information and Event Management (SIEM) solution
  - Quickly detect and respond to internal and external attacks
- Splunk User Behavior Analytics (UBA)

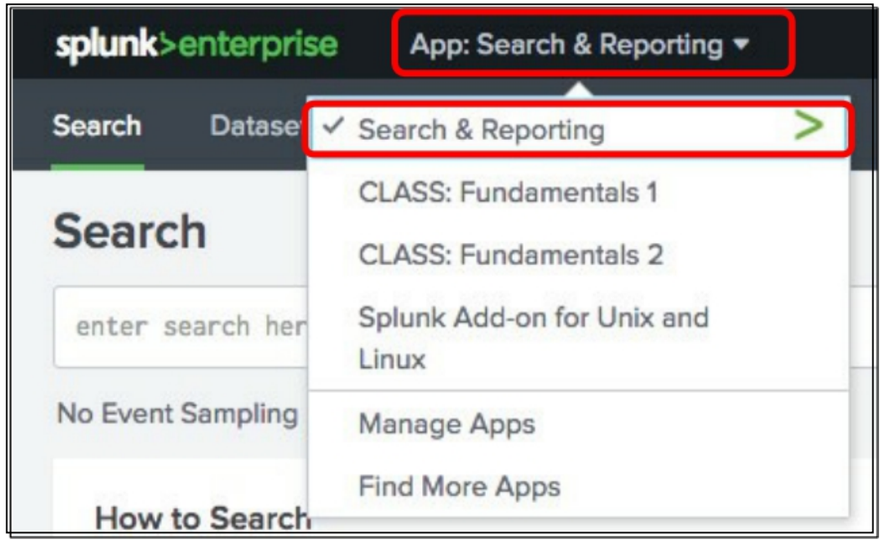
  Finds known, unknown, and hidden threats by analyzing user behavior and flagging unusual activity

Users and Roles

- Splunk users are assigned roles, which determine their capabilities and data access
- Out of the box, there are 3 main roles:
  - Admin
  - Power
  - User
- Splunk admins can create additional roles

Search & Reporting App

- Provides a default interface for searching and analyzing data
- Enables you to create knowledge objects, reports, and dashboards
- Access by selecting the **Search & Reporting** button on the Home app or from an app view, select **Apps** > **Search & Reporting**



Course Scenario

- Use cases in this course are based on Buttercup Games, a fictitious gaming company
- Multinational company with its HQ in San Francisco and offices in Boston and London
- Sells products through its worldwide chain of 3rd party stores and through its online store

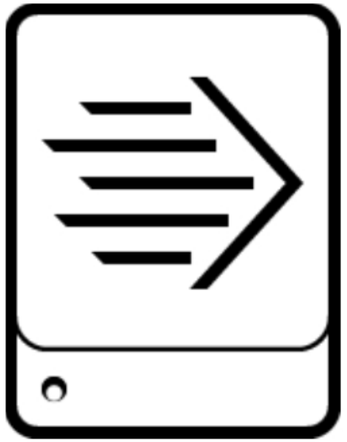
Splunk Components

Splunk Components

Splunk is comprised of three main processing components:

Indexer

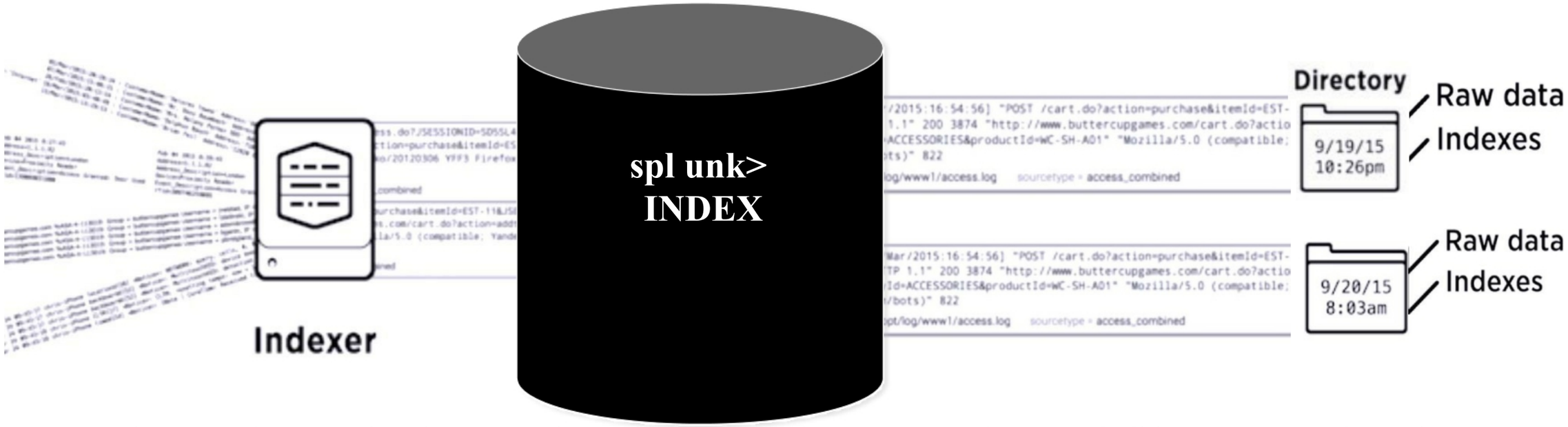
Search Head



Forwarder

Splunk Components - Indexer

· Processes machine data, storing the results in indexes as events, enabling fast search and analysis

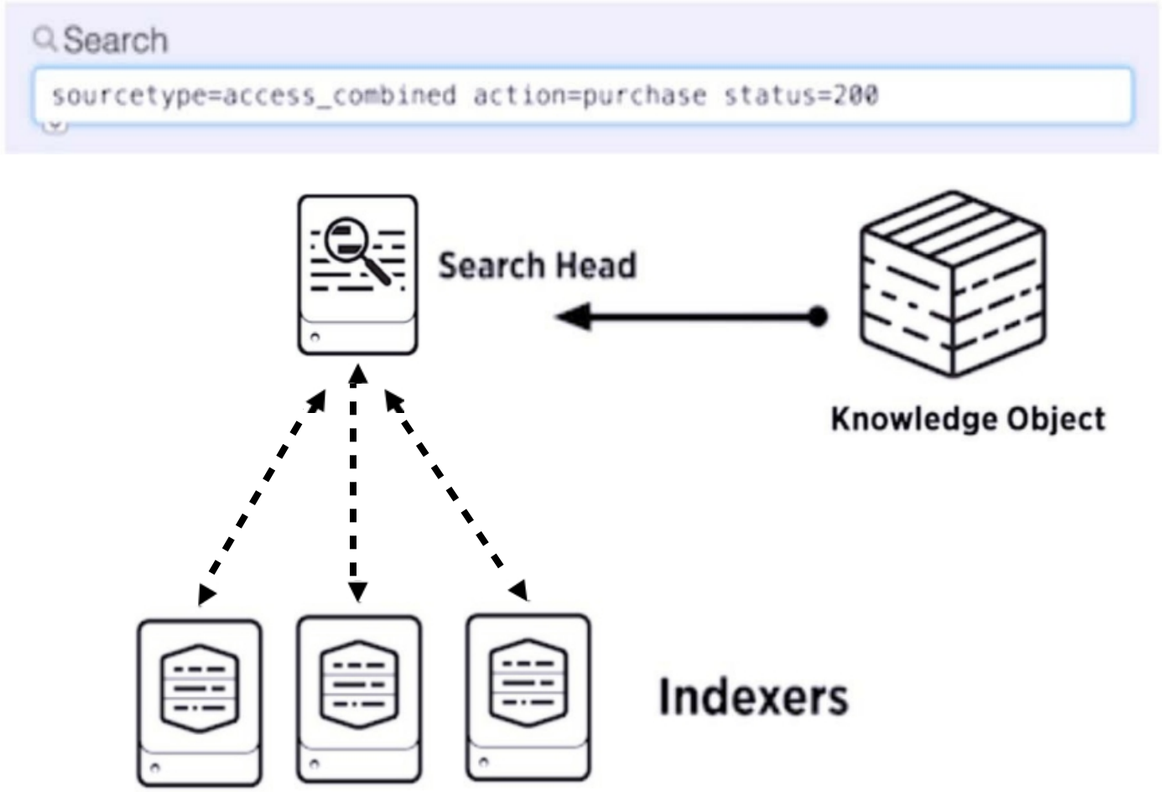


· As the Indexer indexes data, it creates a number of files organized in sets of directories by age

– Contains raw data (compressed) and indexes (points to the raw data)

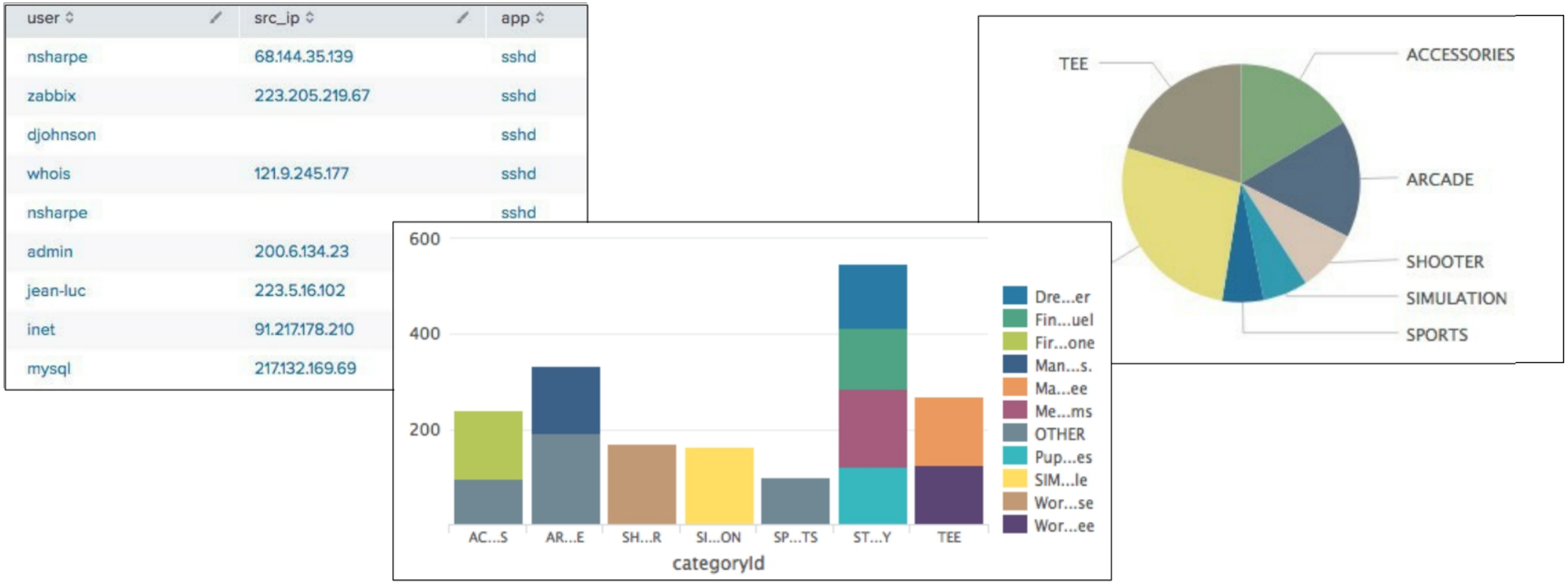
Splunk Components – Search Heads

- Allows users to use the Search language to search the indexed data
- Distributes user search requests to the Indexers
- Consolidates the results and extracts field value pairs from the events to the user
- Knowledge Objects on the Search Heads can be created to extract additional fields and transform the data without changing the underlying index data



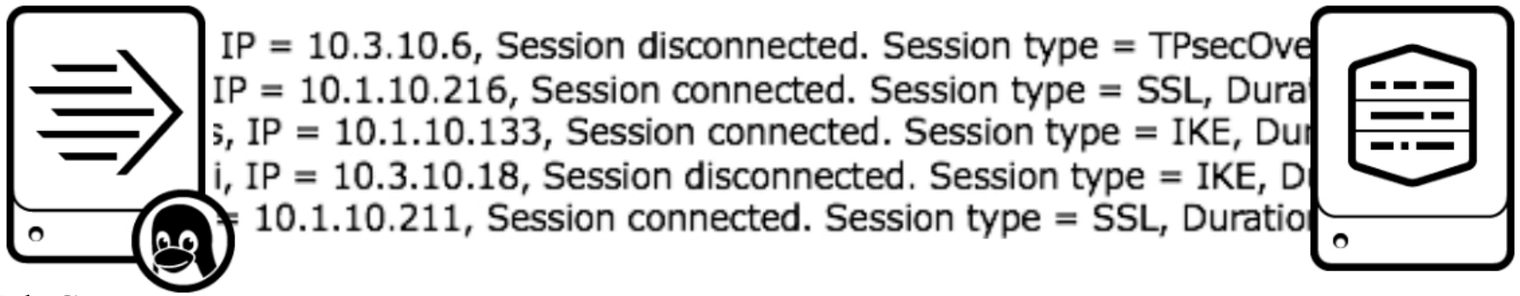
Splunk Components – Search Heads (cont.)

Search Heads also provide tools to enhance the search experience such as reports, dashboards and visualizations



Splunk Components – Forwarders

- Splunk Enterprise instances that consume and send data to the index
- Require minimal resources and have little impact on performance
- Typically reside on the machines where the data originates
- Primary way data is supplied for indexing

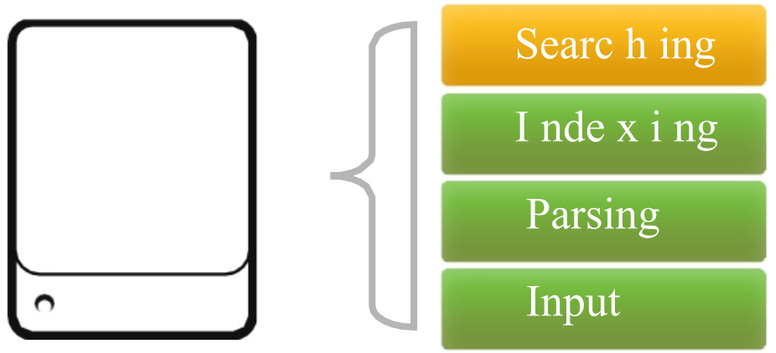


Web Server
with Forwarder instance
i n s t a lle d

I n d e x e r

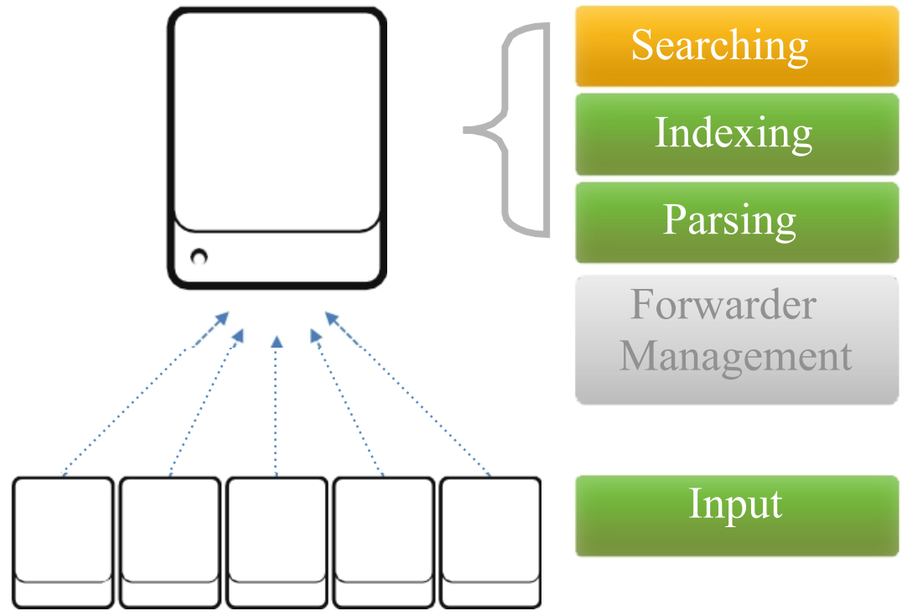
Splunk Deployment – Standalone

- **Single Server**
  - All functions in a single instance of Splunk
  - For testing, proof of concept, personal use, and learning
  - This is what you get when you download Splunk and install with default settings
- Recommendation
  - Have at least one test/ development setup at your site



Splunk Deployment – Basic

- Splunk server
  - Similar to server in standalone configuration
  - Manage deployment of forwarder configurations
- Forwarders
  - Forwarders collect data and send it to Splunk servers
  - Install forwarders at data source (usually production servers)

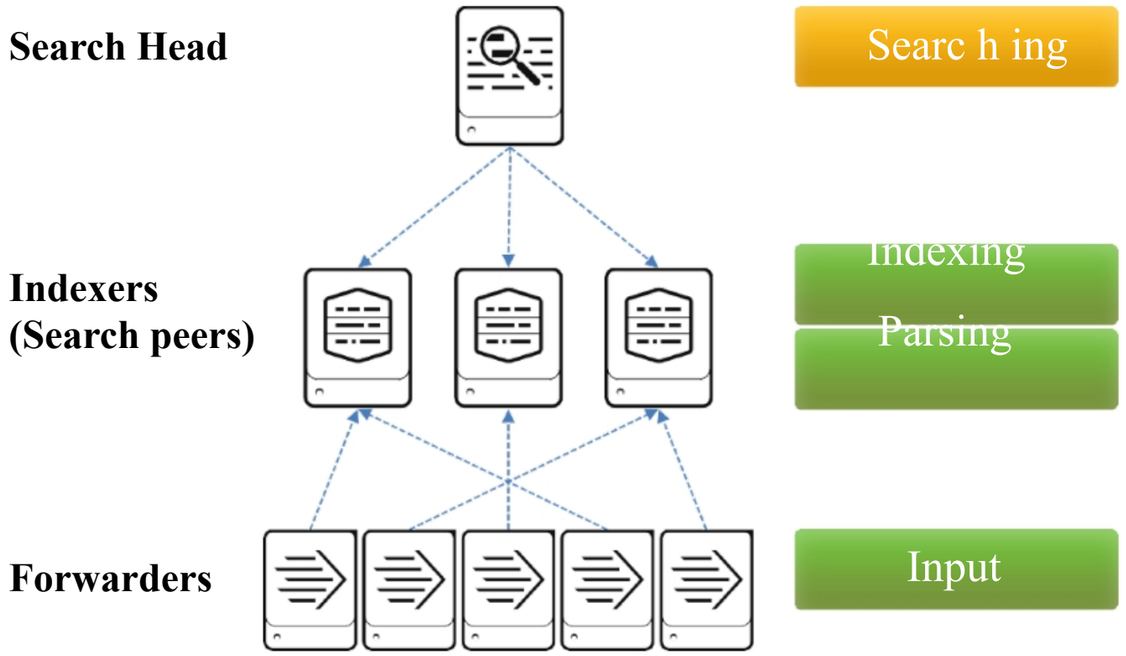


Basic Deployment for organizations:

- Indexing less than 20GB per day
- With under 20 users
- Small amount of forwarders

Splunk Deployment – Multi-Instance

- Increases indexing and searching capacity
- Search management and index functions are split across multiple machines

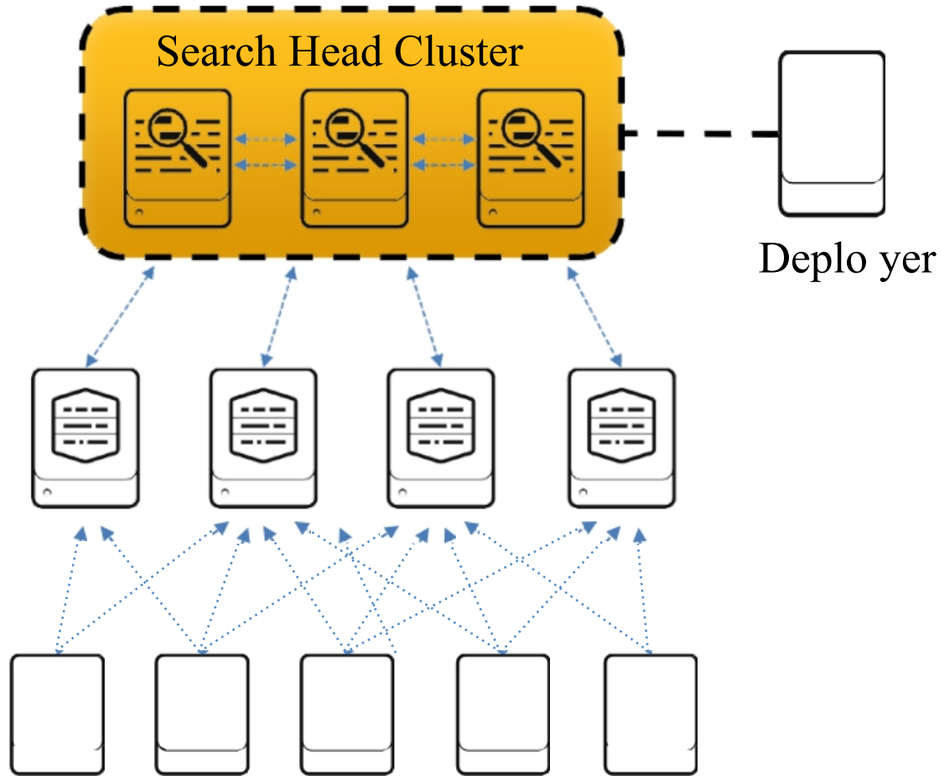


Deployment for organizations:

- Indexing up to 100 GB per day
- Supports 100 users
- Supports several hundred forwarders

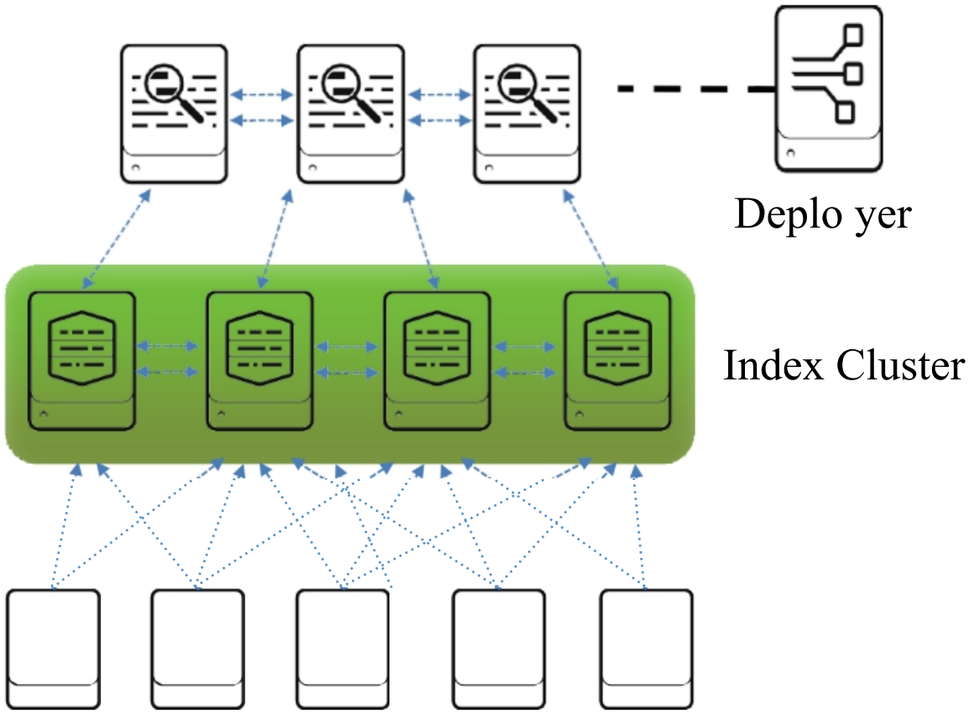
Splunk Deployment – Increasing Capacity

- Adding a Search Head Cluster:
  - Services more users for increased search capacity
  - Allows users and searches to share resources
  - Coordinate activities to handle search requests and distribute the requests across the set of indexers
- Search Head Clusters require a minimum of three Search Heads
- A Deployer is used to manage and distribute apps to the members of the Search Head Cluster



Splunk Deployment – Index Cluster

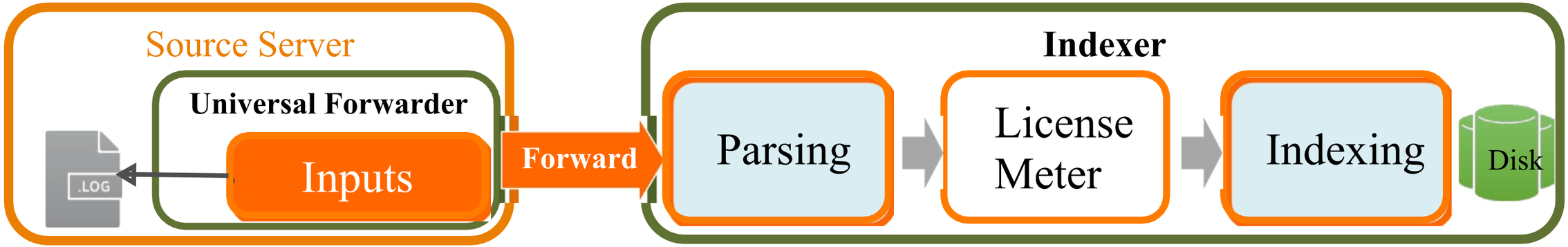
- Traditional Index Clusters:
  - Configured to replicate data
  - Prevent data loss
  - Promote availability
  - Manage multiple indexers
- Non-replicating Index Clusters
  - Offer simplified management
  - Do not provide availability or data recovery



Getting Data In

Splunk Index Time Process

- Splunk index time process (data ingestion) can be broken down into three phases:
  1. **Input phase**: handled at the source (usually a forwarder)
     The data sources are being opened and read
     Data is handled as streams and any configuration settings are applied to the entire stream
  2. **Parsing phase**: handled by indexers (or heavy forwarders)
     Data is broken up into events and advanced processing can be performed
  3. **Indexing phase**:
     License meter runs as data and is initially written to disk, prior to compression
     After data is written to disk, it **cannot** be changed



Data Input Types

- Splunk supports many types of data input
  - **Files and directories:** monitoring text files and/or directory structures containing text files
  - **Network data:** listening on a port for network data
  - **Script output:** executing a script and using the output from the script as the input
  - **Windows logs:** monitoring Windows event logs, Active Directory, etc.
  - **HTTP**: using the HTTP Event Collector
  - And more...
- You can add data inputs with:
  - Apps and add-ons from Splunkbase
  - Splunk Web
  - CLI
  - Directly editing **`inputs.conf`**

Default Metadata Settings

- When you index a data source, Splunk assigns metadata values
  - The metadata is applied to the entire source
  - Splunk applies defaults if not specified
  - You can also override them at input time or later

Metadata	Default
`source`	Path of input file, network hostname:port, or script name
`host`	Splunk hostname of the inputting instance (usually a forwarder)
`sourcetype`	Uses the source filename if Splunk cannot automatically determine
`index`	Defaults to **`main`**

Adding an Input with Splunk Web

- Splunk admins have a number of ways to start the **Add Data** page
  - Click the **Add Data** icon

    On the admin's **Home** page

    On the **Settings** panel
  - Select **Settings** > **Data inputs** > **Add new**

Set Source Type

Splunk automatically determines the source type for major data types when there is enough data

You can choose a different source type from the dropdown list

Or, you can create a new source type name for the specific source

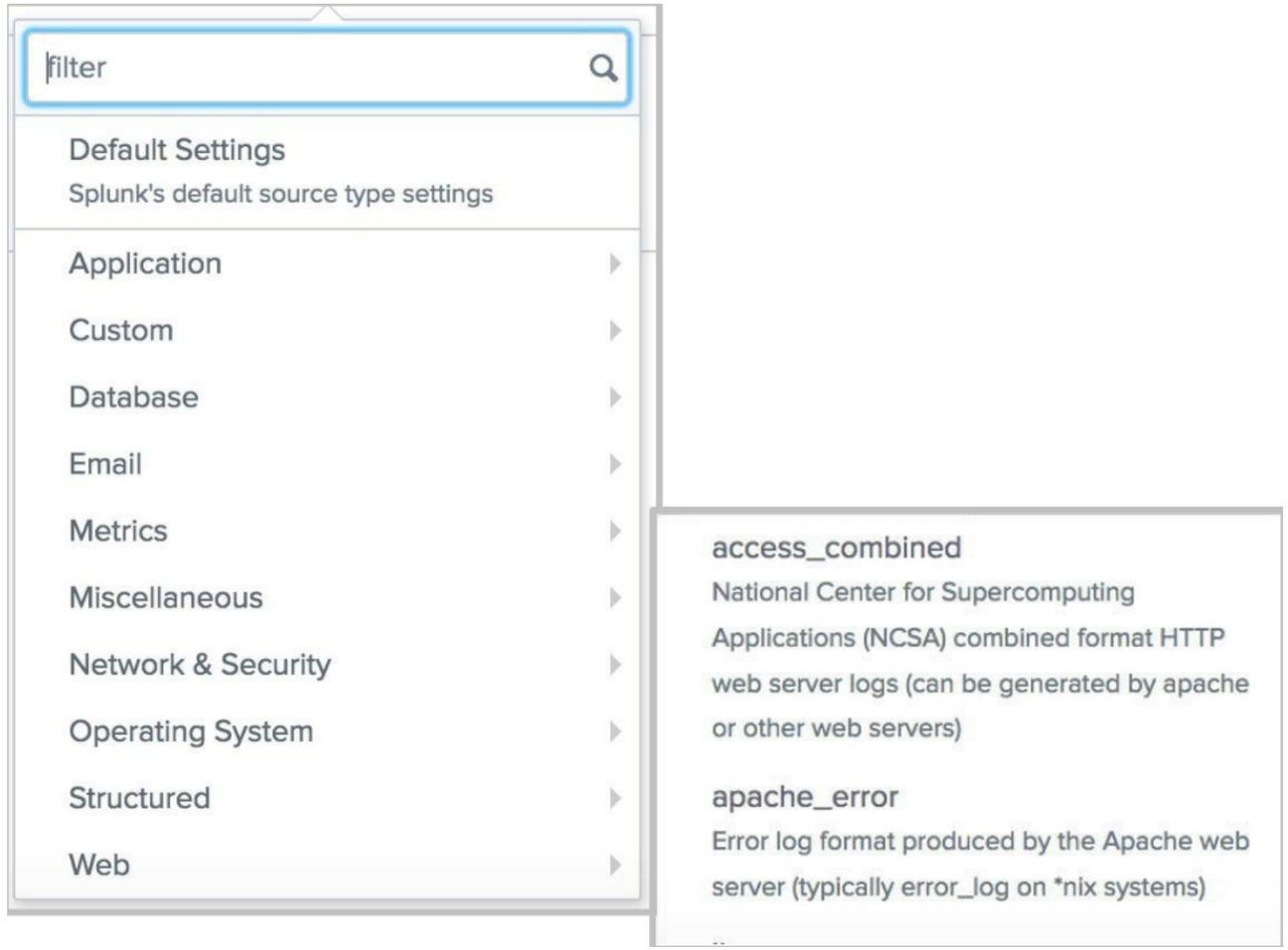
Data preview displays how your processed events will be indexed

- If the events are correctly separated and the right timestamps are highlighted, you can move ahead

  If not, you can select a different source type from the list or customize the settings

Pretrained Source Types

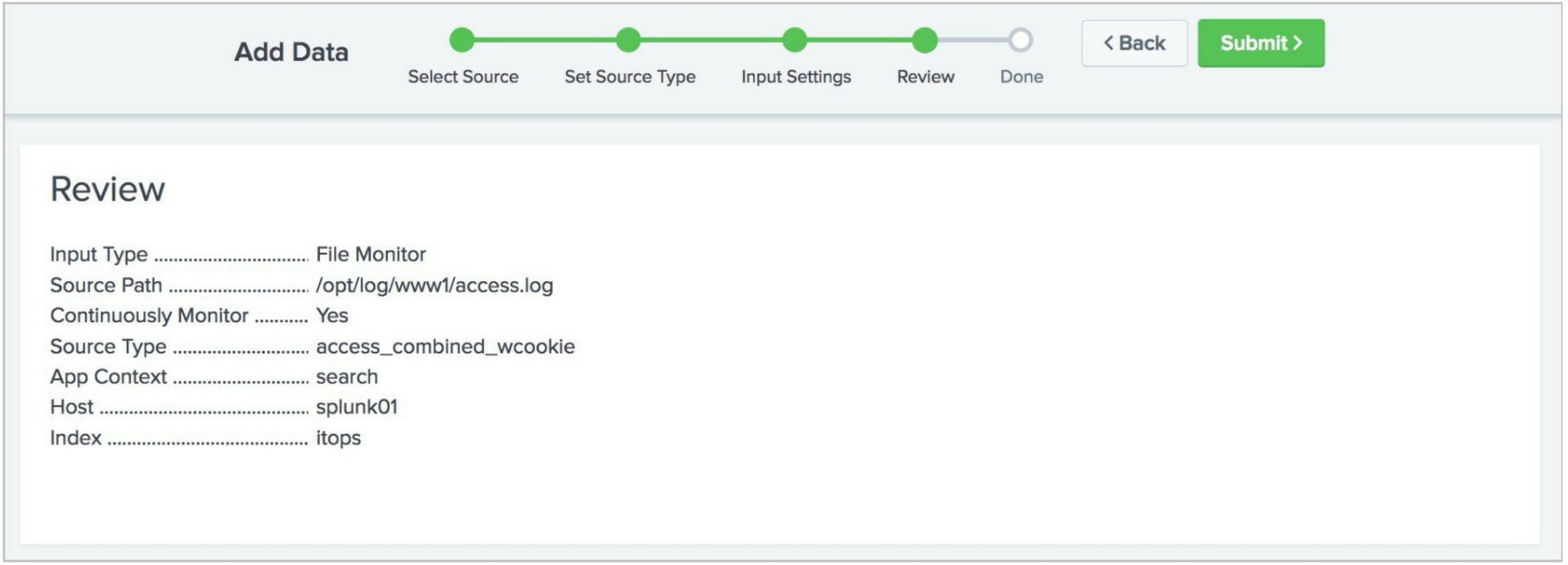
- Splunk has default settings for many types of data
- The docs also contain a list of source types that Splunk automatically recognizes
- Splunk apps can be used to define additional source types



http://docs.splunk.com/Documentation/Splunk/latest/Data/Listofpretrainedsourcetypes

Review

· Review the input configuration summary and click **Submit** to finalize



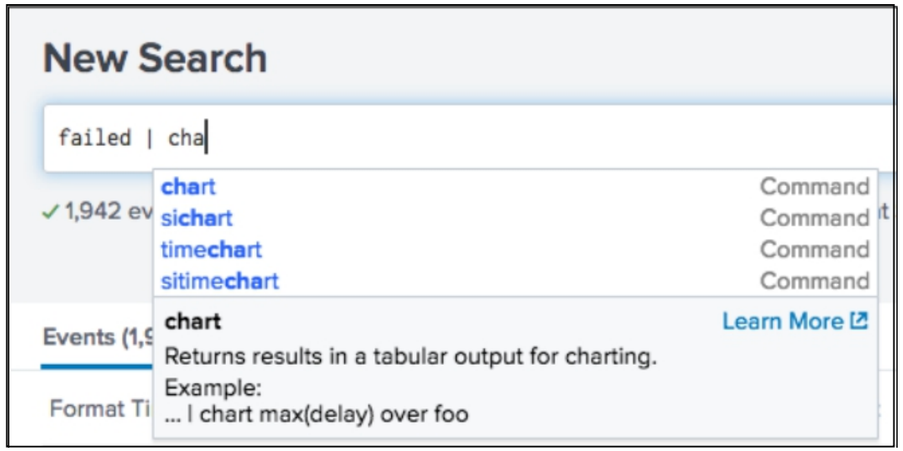
Module 5:
Basic Search

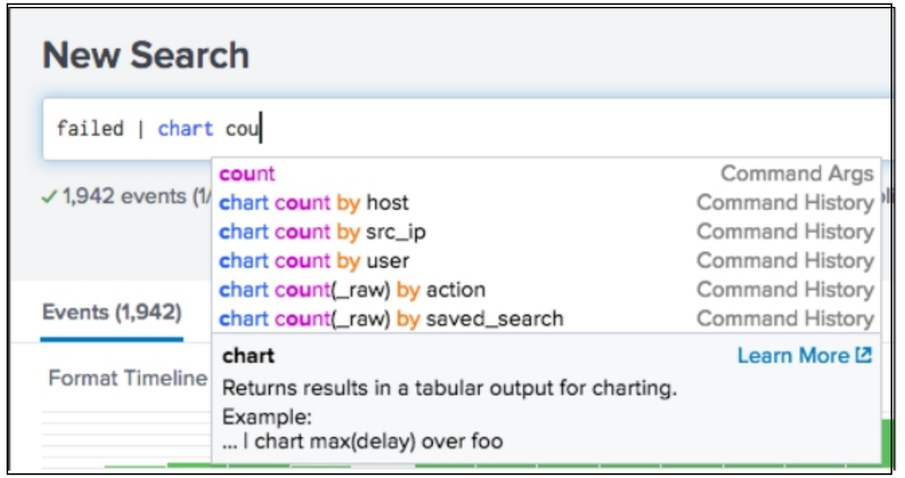
Search Assistant

- Search Assistant provides selections for how to complete the search string
- Before the first pipe (|), it looks for matching terms
- You can continue typing OR select a term from the list
  - If you select a term from the list, it is added to the search

Search Assistant (cont.)

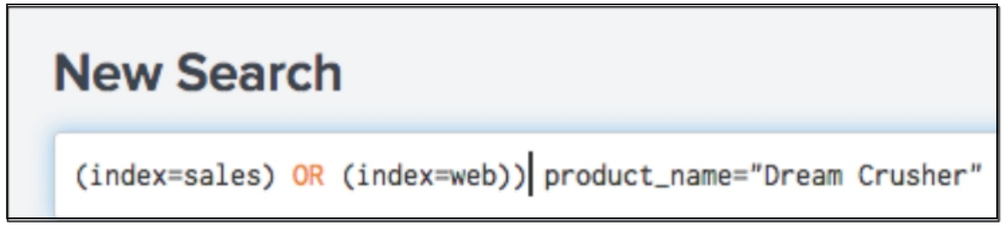
- After the first pipe, the Search Assistant shows a list of commands that can be entered into the search string
- You can continue typing OR scroll through and select a command to add
- If you mouse over a command, more information about the command is shown
- As you continue to type, Search Assistant makes more suggestions





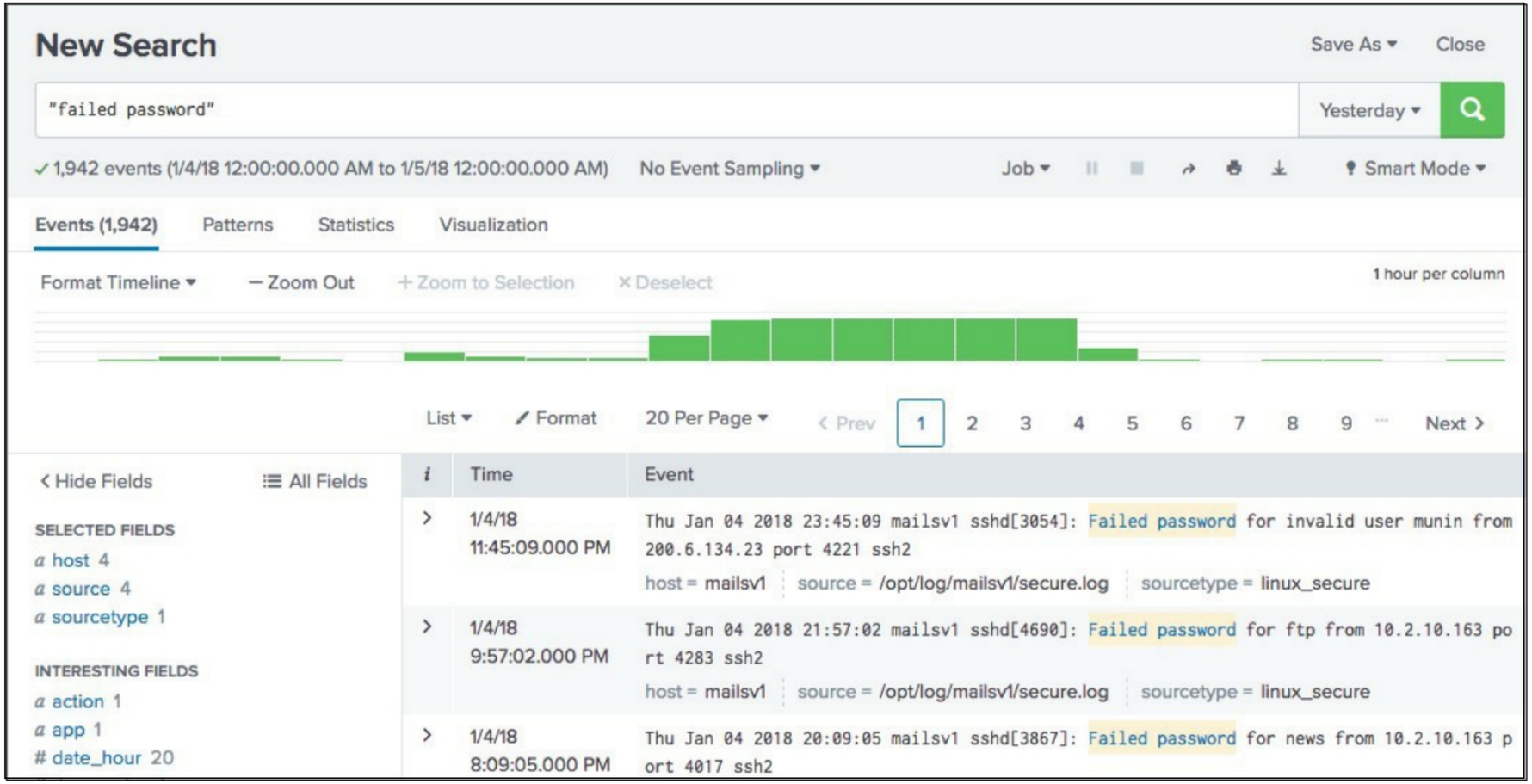
Search Assistant – Parentheses

- The Search Assistant provides help to match parentheses as you type
- When an end parenthesis is typed, the corresponding beginning parenthesis is automatically highlighted
  - If a beginning parenthesis cannot be found, *nothing* is highlighted



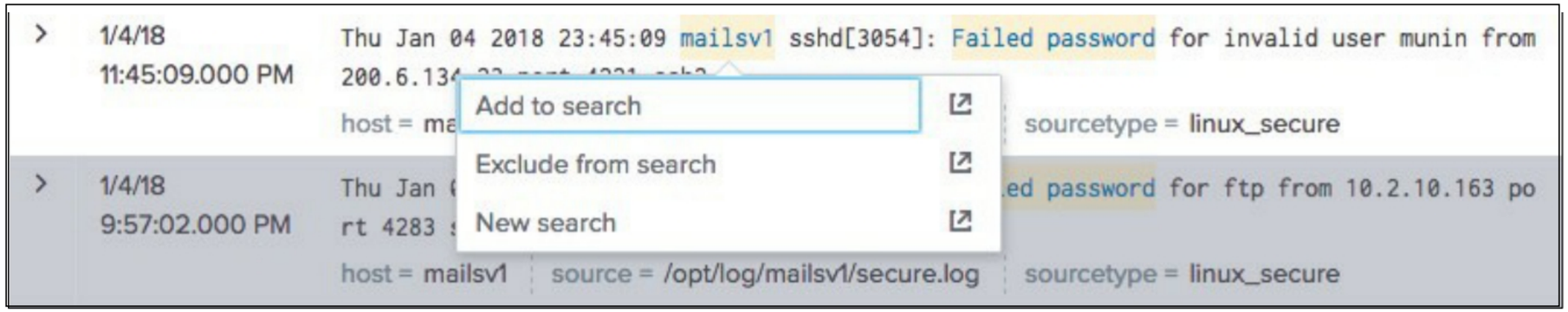
Viewing Search Results

- Matching results are returned immediately
- Displayed in reverse chronological order (newest first)
- Matching search terms are highlighted



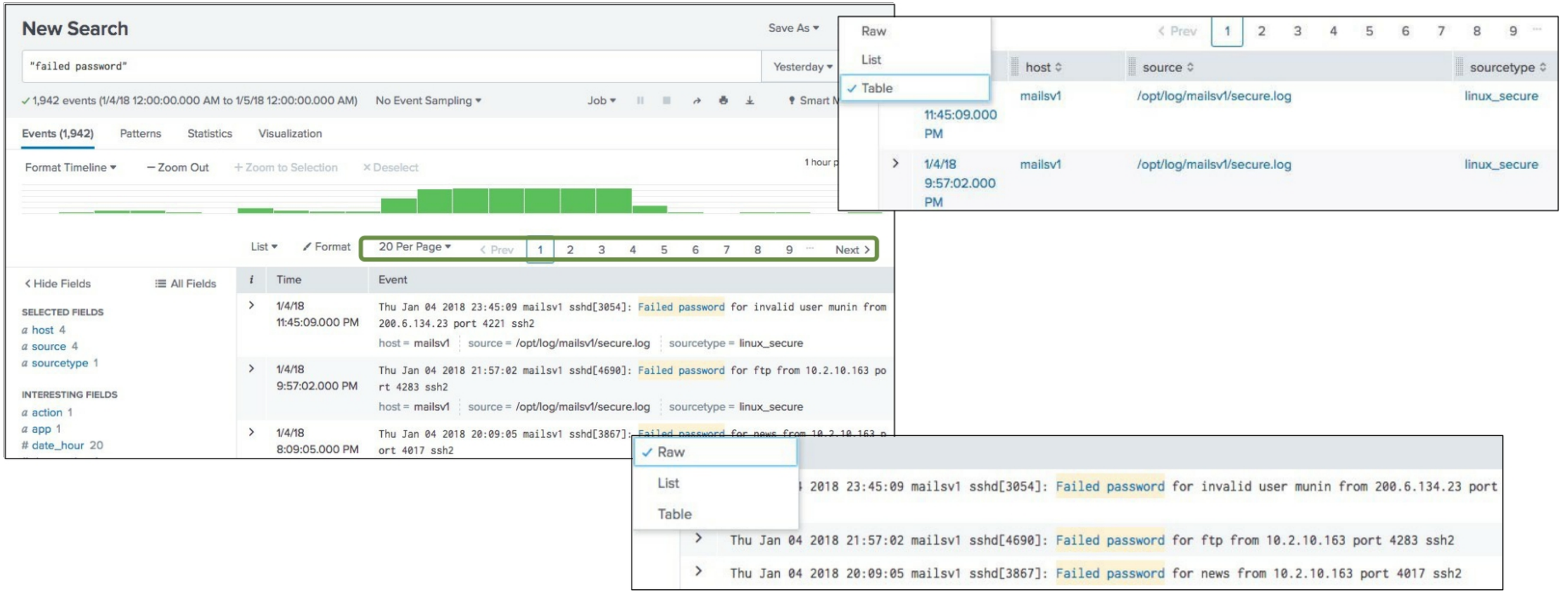
Using Search Results to Modify a Search

- When you mouse over search results, keywords are highlighted
- Click any item in your search results; a window appears allowing you to:
  - Add the item to the search
  - Exclude the item from the search
  - Open a new search including only that item



Changing Search Results View Options

You have several layout options for displaying your search results



Time Range Abbreviations

- Time ranges are specified in the **Advanced** tab of the time range picker
- Time unit abbreviations include:

s = seconds m = minutes h = hours d = days w = week mon = months y = year

- **@** symbol "snaps " to the time unit you specify
  - Snapping rounds *down* to the nearest specified unit
  - Example: Current time when the search starts is 09:37:12

    -30m@h looks back to 09:00:00

Time Range: earliest and latest

· You can also specify a time range in the search bar

· To specify a beginning and an ending for a time range, use earliest and latest

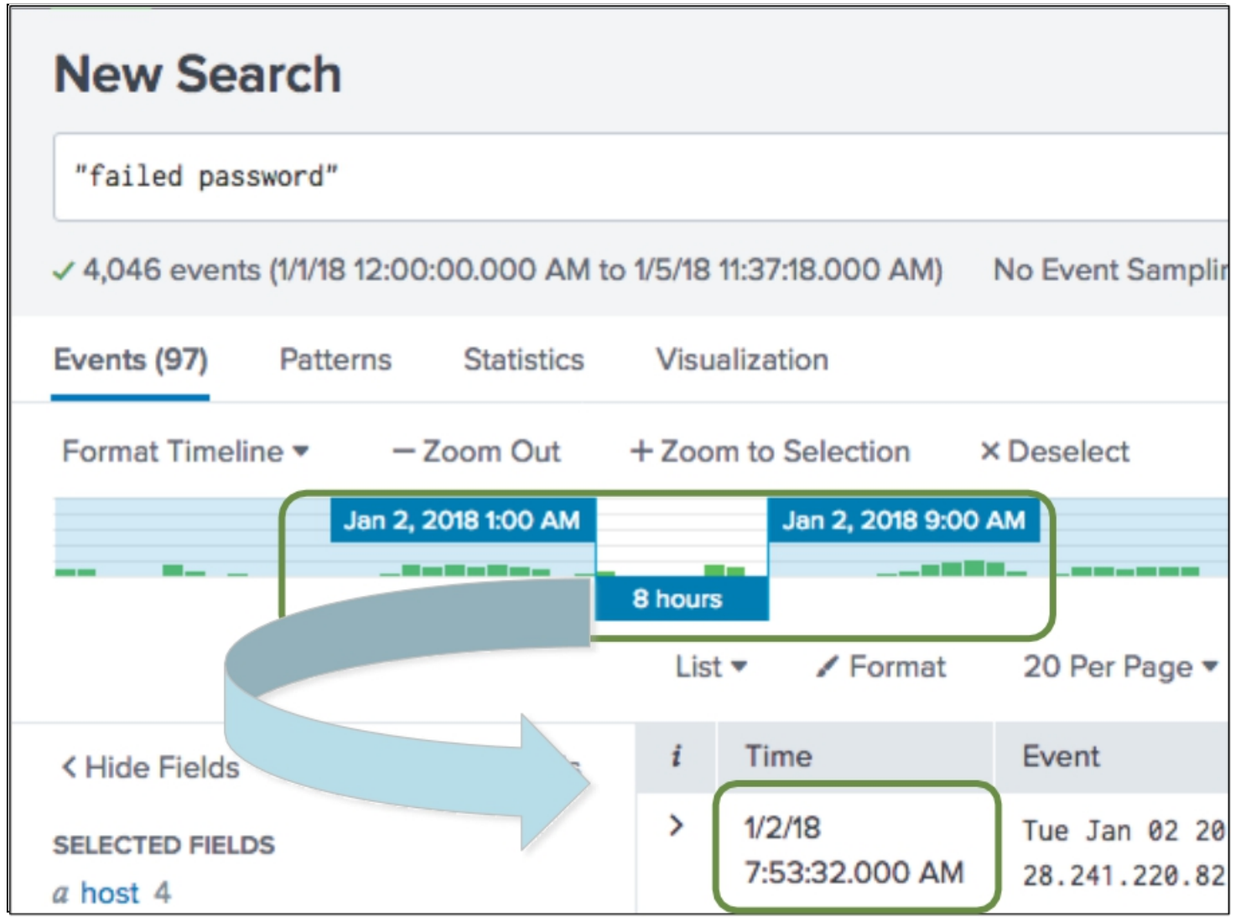
· Examples:

earliest=-h	looks back one hour
earliest=-2d@ latest=@d	looks back from two days ago, up to the beginning of today
earliest=6/15/2017:12:30:00	looks back to specified time

Viewing a Subset of the Results with Timeline

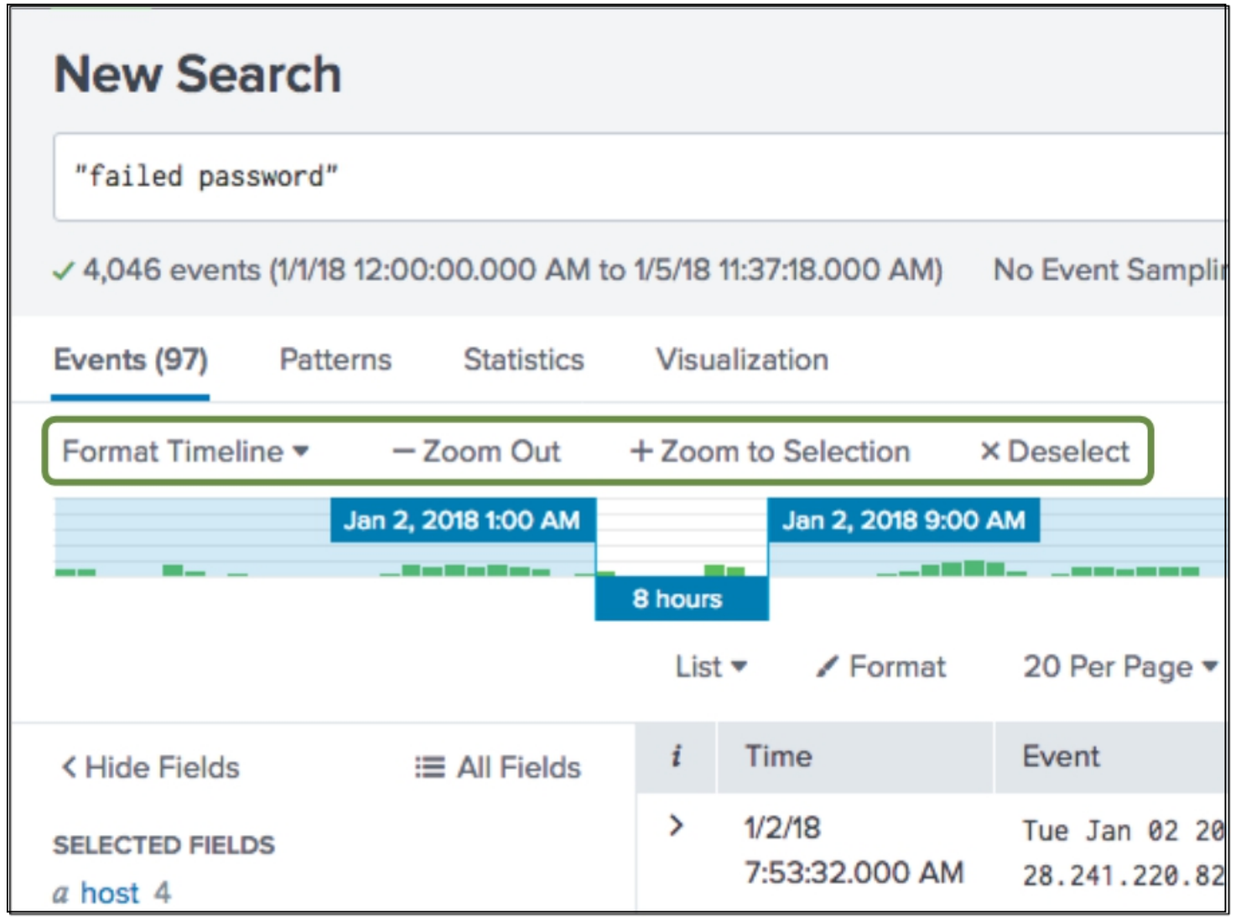
- To select a narrower time range, click and drag across a series of bars
  - This action filters the current search results

    Does not re-execute the search
  - This filters the events and displays them in reverse chronological order (most recent first)



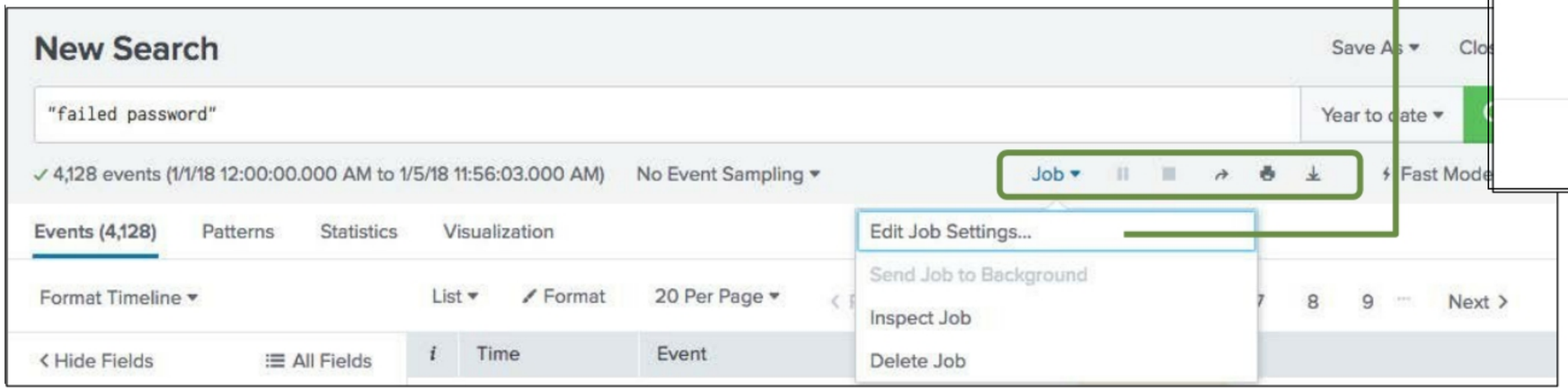
Using Other Timeline Controls

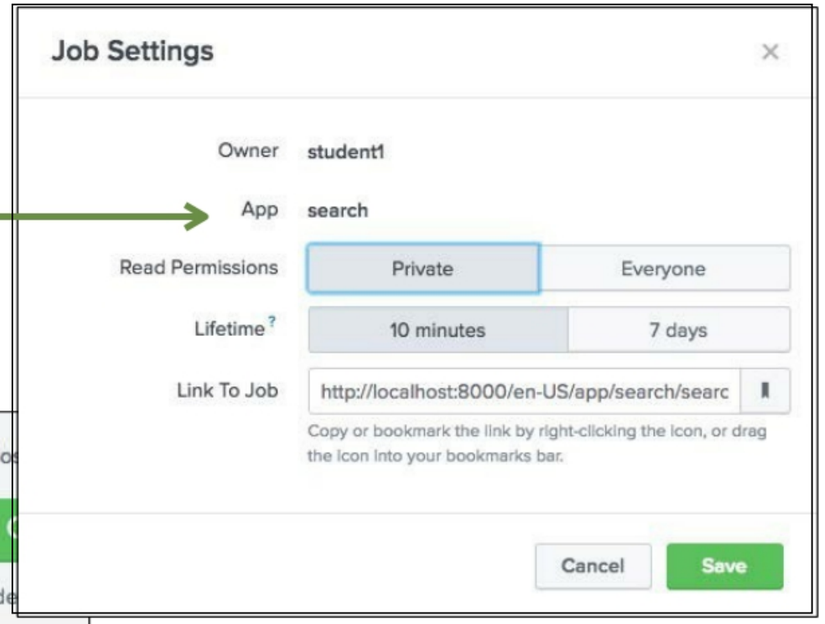
- **Format Timeline**
  - Hides or shows the timeline in different views
- **Zoom Out**
  - Expands the time focus and re-executes the search
- **Zoom to Selection**
  - Narrows the time range and re-executes the search
- **Deselect**
  - If in a drilldown, returns to the original results set
  - Otherwise, grayed out / unavailable



Controlling and Saving Search Jobs

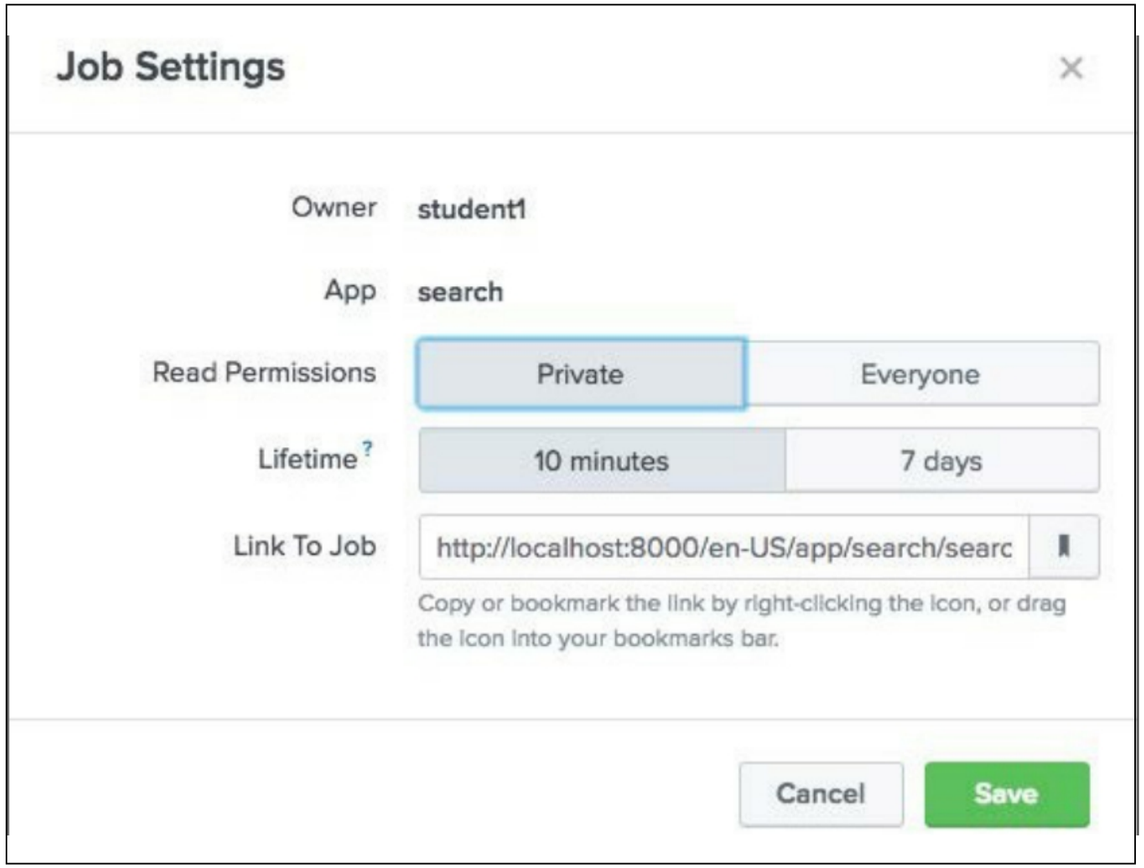
- Every search is also a **job**
- Use the Job bar to control search execution
  - **Pause** – toggles to resume the search
  - **Stop** – finalizes the search in progress
    - Jobs are available for 10 minutes (default)
    - Get a link to results from the **Job** menu





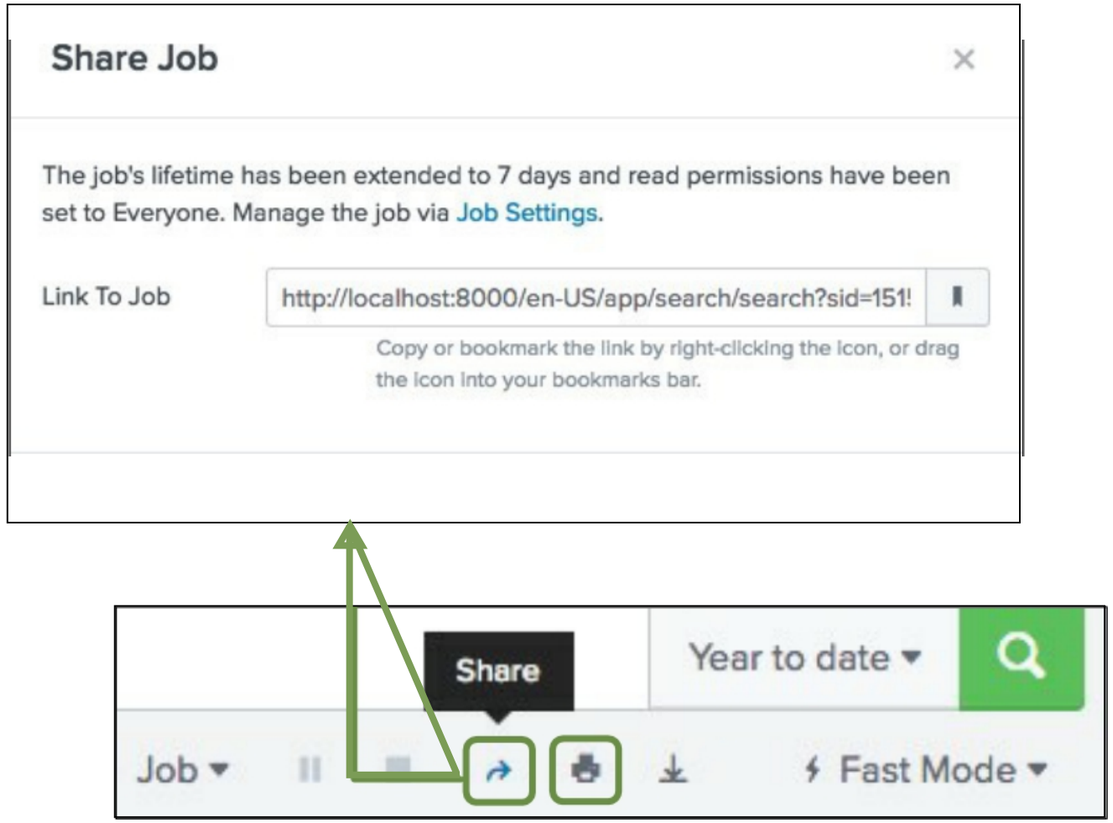
Setting Permissions

- **Private** [default]
  - Only the creator can access
- **Everyone**
  - All app users can access search results
- **Lifetime**
  - Default is 10 minutes
  - Can be extended to 7 days
  - To keep your search results longer, schedule a report



Sharing Search Jobs

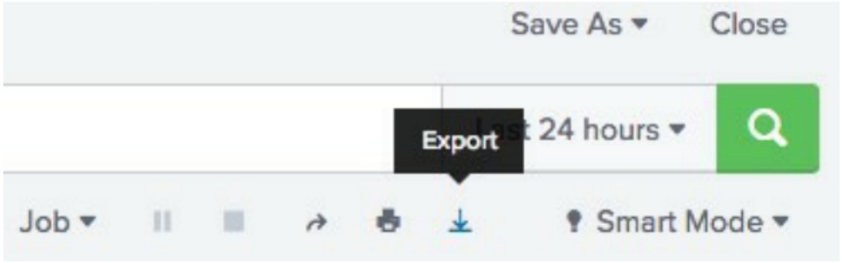
- Use the Share button next to the Job bar to quickly:
  - Give everyone read permissions
  - Extend results retention to 7 days
  - Get a sharable link to the results
- Sharing search allows multiple users working on same issue to see same data
  - More efficient than each running search separately
  - Less load on server and disk space used

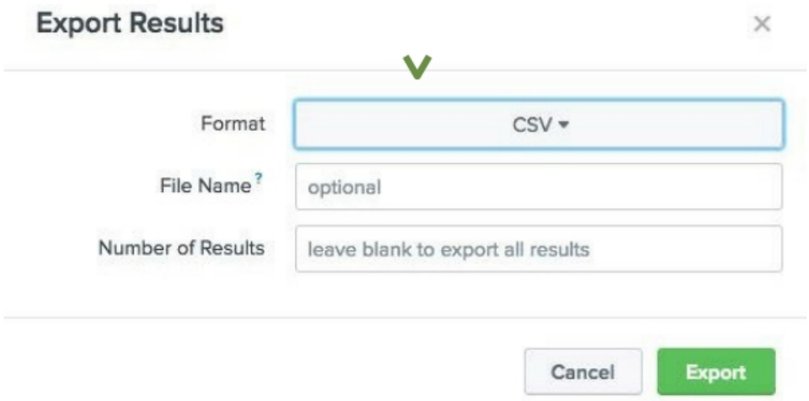


- Can also click printer icon to print results or save as PDF

Exporting Search Results

For an external copy of the results, **export** search results to Raw Events (text file), CSV, XML, or JSON format





Viewing Your Saved Jobs

- Access saved search jobs from the **Activity** menu
- The Search Jobs view displays jobs that:
  - You have run in the last 10 minutes
  - You have extended for 7 days
- Click on a job link to view the results in the designated app view

Viewing Your Search History

1. Search History displays your most recent ad-hoc searches – 5 per page
2. You can set a time filter to further narrow your results
3. Click the > icon in the leftmost column to expand long queries to display the full text

Using Fields in Searches

What Are Fields?

- Fields are searchable key/value pairs in your event data
  - Examples: host=www1 status=503
- Fields can be searched with their names, like separating an http status code of 404 from Atlanta's area code (area_code=404)
- Between search terms, AND is implied unless otherwise specified

Field Discovery

- Splunk automatically discovers many fields based on sourcetype and key/value pairs found in the data
- Prior to search time, some fields are already stored with the event in the index:
  - Meta fields, such as **host**, **source**, **sourcetype**, and **index**
  - Internal fields such as **_time** and **_raw**
- At search time, *field discovery* discovers fields directly related to the search's results
- Some fields in the overall data may not appear within the results of a particular search

Identify Data-Specific Fields

- Data-specific fields come from the specific characteristics of your data
  - Sometimes, this is indicated by obvious key = value pairs (**action = purchase**)
  - Sometimes, this comes from data within the event, defined by the sourcetype (**status = 200**)

i	Time	Event
>	1/5/18 1:21:10.000 PM	192.162.19.179 - - [05/Jan/2018:13:21:10] "POST /cart/success.do?JSESSIONID=SD1SL6FF4ADFF4964 HTTP 1.1" 200 966 "http://www.buttercupgames.com/cart.do?action=purchase&itemId=EST-26" "Mozilla/5.0 (iPad; U; CPU OS 4_3_5 like Mac OS X; en-us) AppleWebKit/533.17.9 (KHTML, like Gecko) Version/5.0.2 Mobile/8L1 Safari/6533.18.5" 552

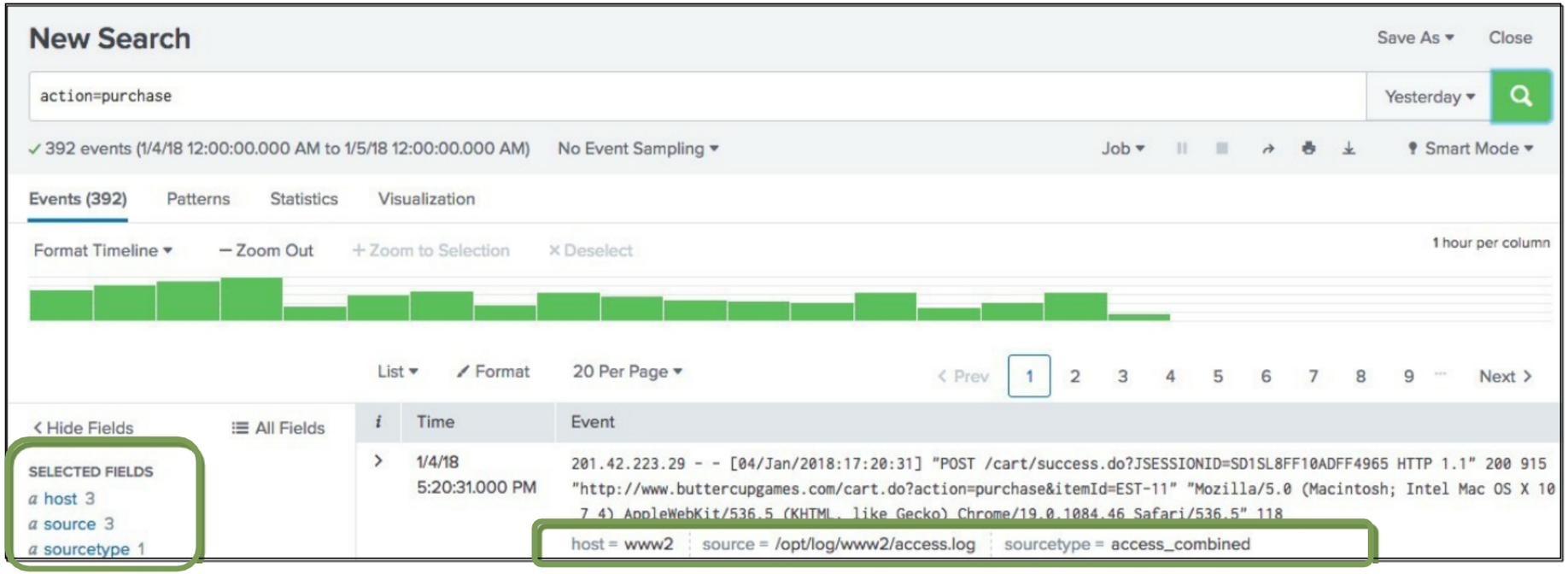
Fields Sidebar

For the current search:

- **Selected Fields** – a set of configurable fields displayed for each event
- **Interesting Fields** – occur in at least 20% of resulting events
- **All Fields** link to view all fields (including non-interesting fields)

Selected Fields

- Selected fields and their values are listed under every event that includes those fields
- By default, the selected fields are:
  - host
  - source
  - sourcetype
- You can choose any field and make it a selected field

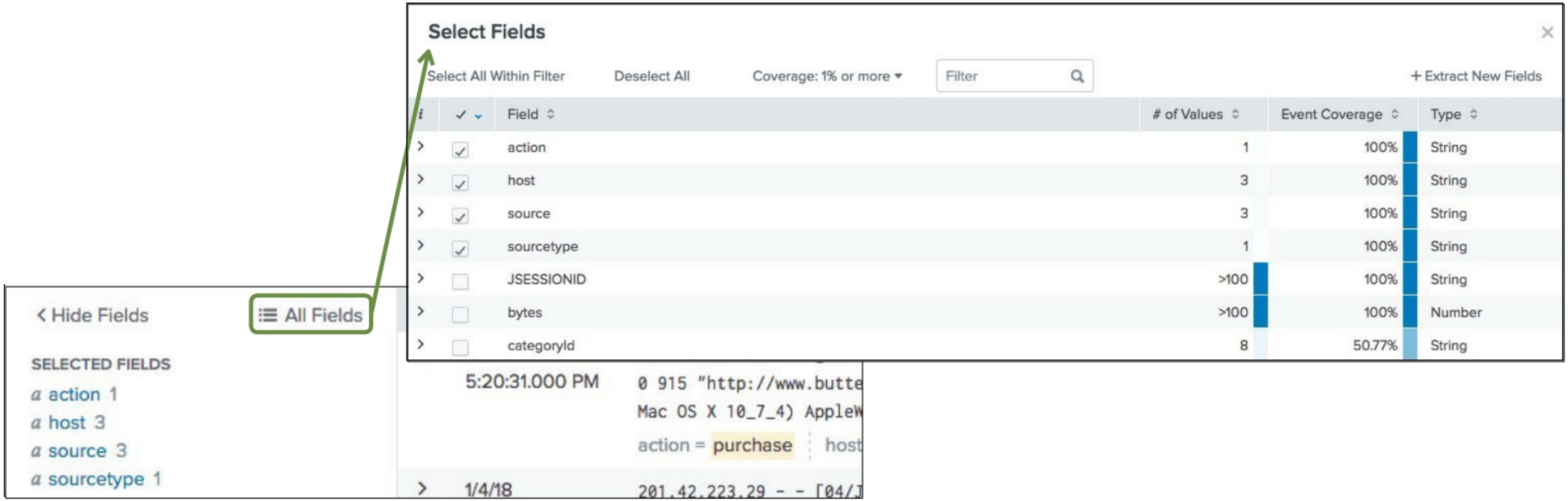


Make an Interesting Field a Selected Field

- You can modify selected fields
  - Click a field in the Fields sidebar
  - Click **Yes** in the upper right of the field dialog
- Note that a selected field appears:
  - In the Selected Fields section of the Fields sidebar
  - Below each event where a value exists for that field

Make Any Field Selected

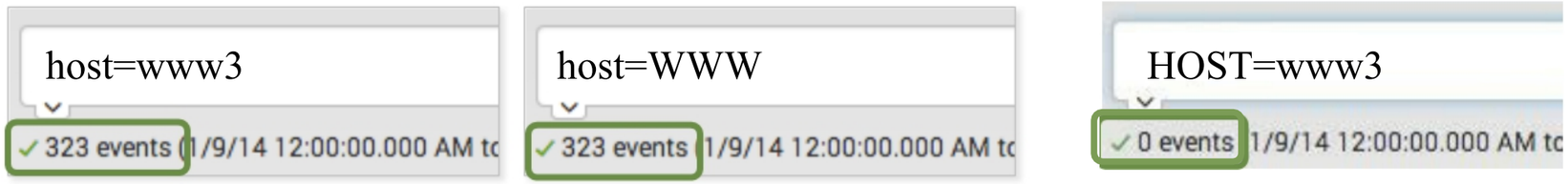
You can identify other fields as selected fields from All Fields (which shows all of the discovered fields)



Using Fields in Searches

- Efficient way to pinpoint searches and refine results

- Field names ARE case sensitive; field values are NOT
  - Example:



Using Fields in Searches (cont.)

- For IP fields, Splunk is subnet/CIDR aware

```
clientip="202.201.1.0/24"
```

```
clientip="202.201.1.*"
```

- Use wildcards to match a range of field values
  - Example: **user=*** (to display all events that contain a value for user)

```
user=*  sourcetype=access*  (referer_domain=*.cn  OR
referer_domain=*.hk)
```

- Use relational operators

With numeric fields

```
src_port>1000
src_port<4000
```

With alphanumeric fields

```
host! =www3
```

!= vs. NOT

· Both!= field expression and NOT operator exclude events from your search, but produce different results

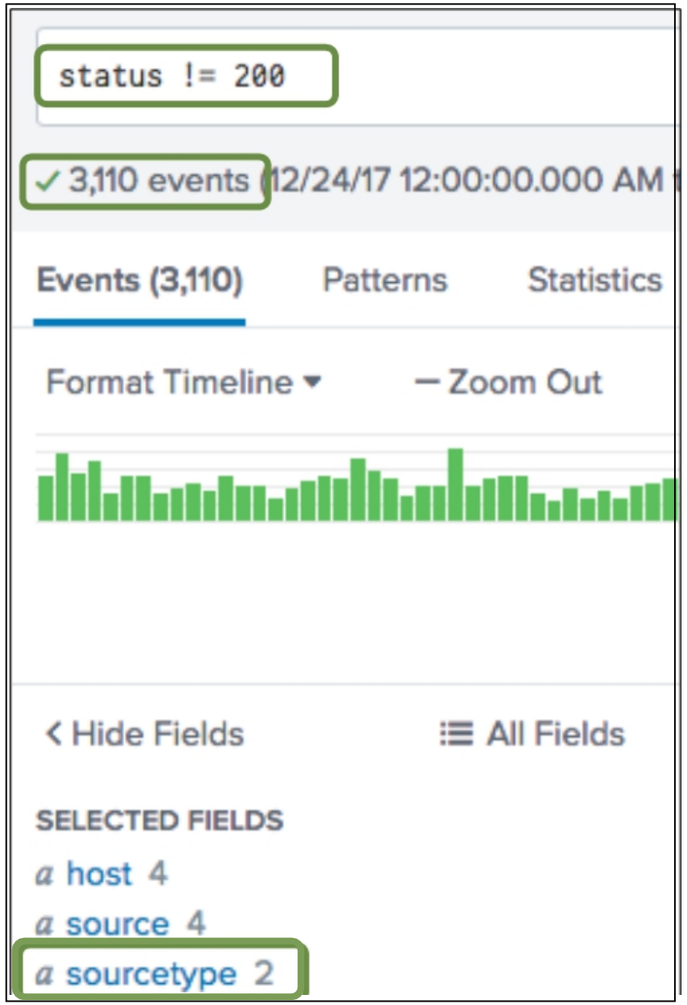
· Example: status != 200

  - Returns events where status field exists and value in field doesn't equal 200

· Example: NOT status = 200

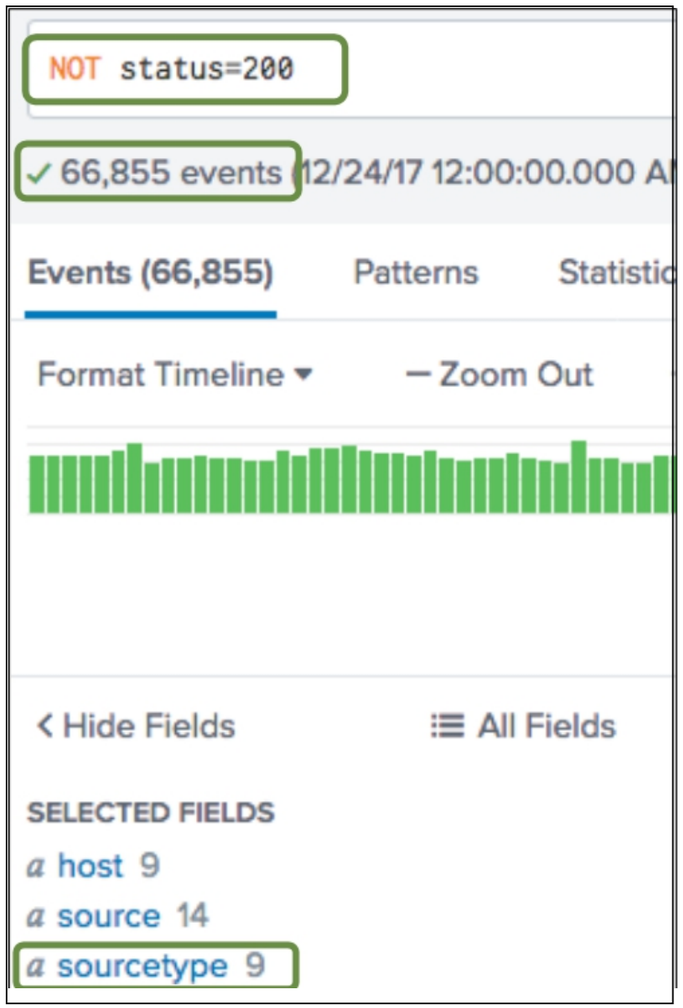
  - Returns events where status field exists and value in field doesn't equal 200 **-- and** all events where status field **doesn't** exist

!= vs. NOT (co nt.)



In this example:

- status != 200 returns **3,110** events from **2** sourcetypes
- NOT status=200 returns **66,855** events from **9** sourcetypes



!= vs. NOT (cont.)

- Does != and NOT ever yield the same results?
  - Yes, if you know the field you're evaluating always exists in the data you're searching
  - For example:

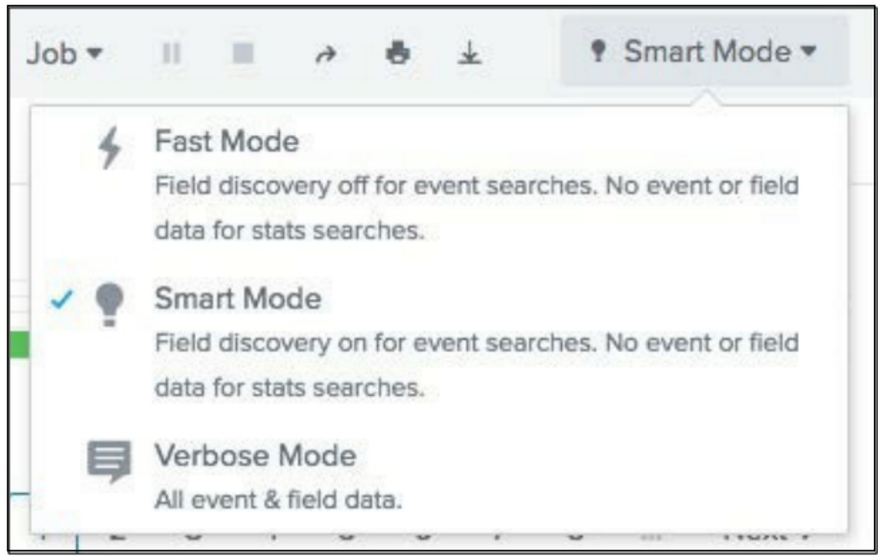
    index=web sourcetype=access_combined status!=200

    index=web sourcetype=access_combined NOT status=200

    yields same results because status field always exists in access_combined s ourc ety pe

Search Modes: Fast, Smart, Verbose

- Fast: emphasizes speed over completeness
- Smart: balances speed and completeness (default)
- Verbose:
  - Emphasizes completeness over speed
  - Allows access to underlying events when using reporting or statistical commands (in addition to totals and stats)



Best Practices

Search Best Practices

- Time is the most efficient filter
- Specify one or more index values at the beginning of your search string
- Include as many search terms as possible

  If you want to find events with "error" and "sshd", and 90% of the events include "error" but only 5% "sshd", include both values in the search
- Make your search terms as specific as possible

  Searching for "access denied" is always better than searching for "denied"
- Inclusion is generally better than exclusion

  Searching for "access denied" is faster than searching for NOT “access granted"

Search Best Practices (cont.)

- Filter as early as possible

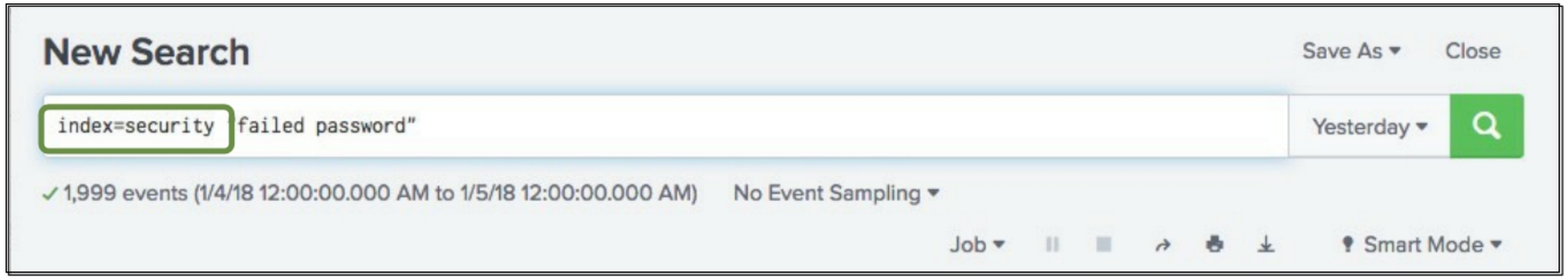
  For example, remove duplicate events, then sort

- Avoid using wildcards at the beginning or middle of a string
  - Wildcards at *beginning* of string scan all events within timeframe
  - Wildcards in *middle* of string may return inconsistent results
  - So use fail* (not *fail or *fail* or f*il)

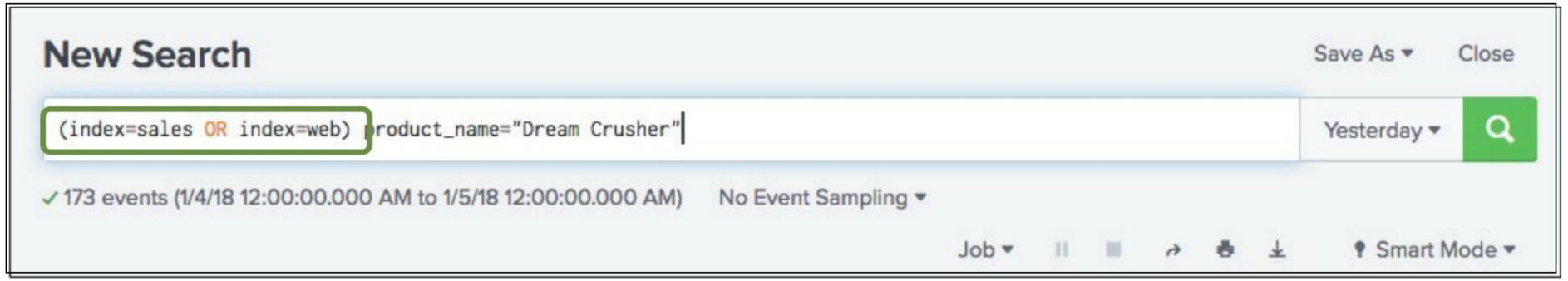
- When possible, use OR instead of wildcards
  - For example, use (user=admin OR user=administrator) instead of user=admin*

Working with Indexes

- This search returns event data from the security index

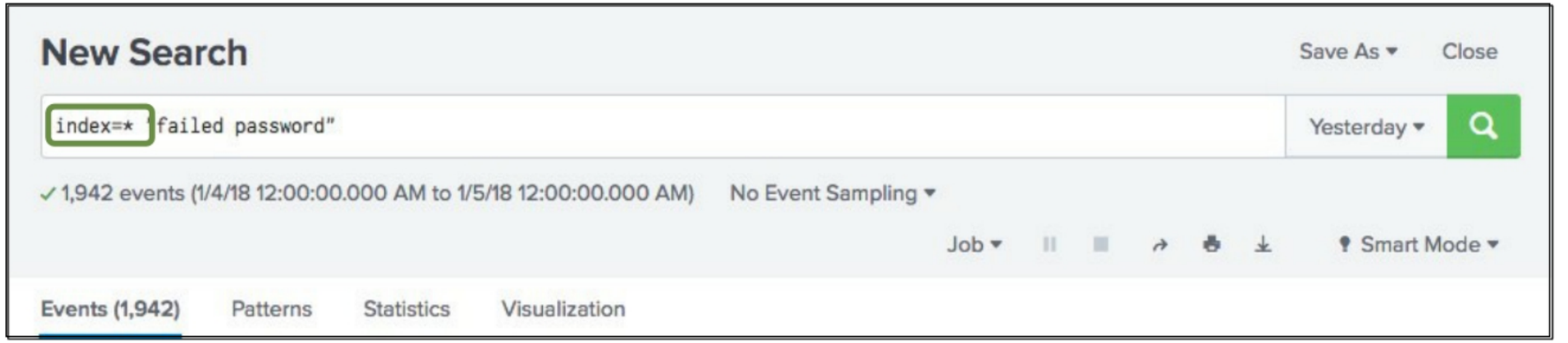


- It's possible to specify multiple index values in a search



Working with Indexes (cont.)

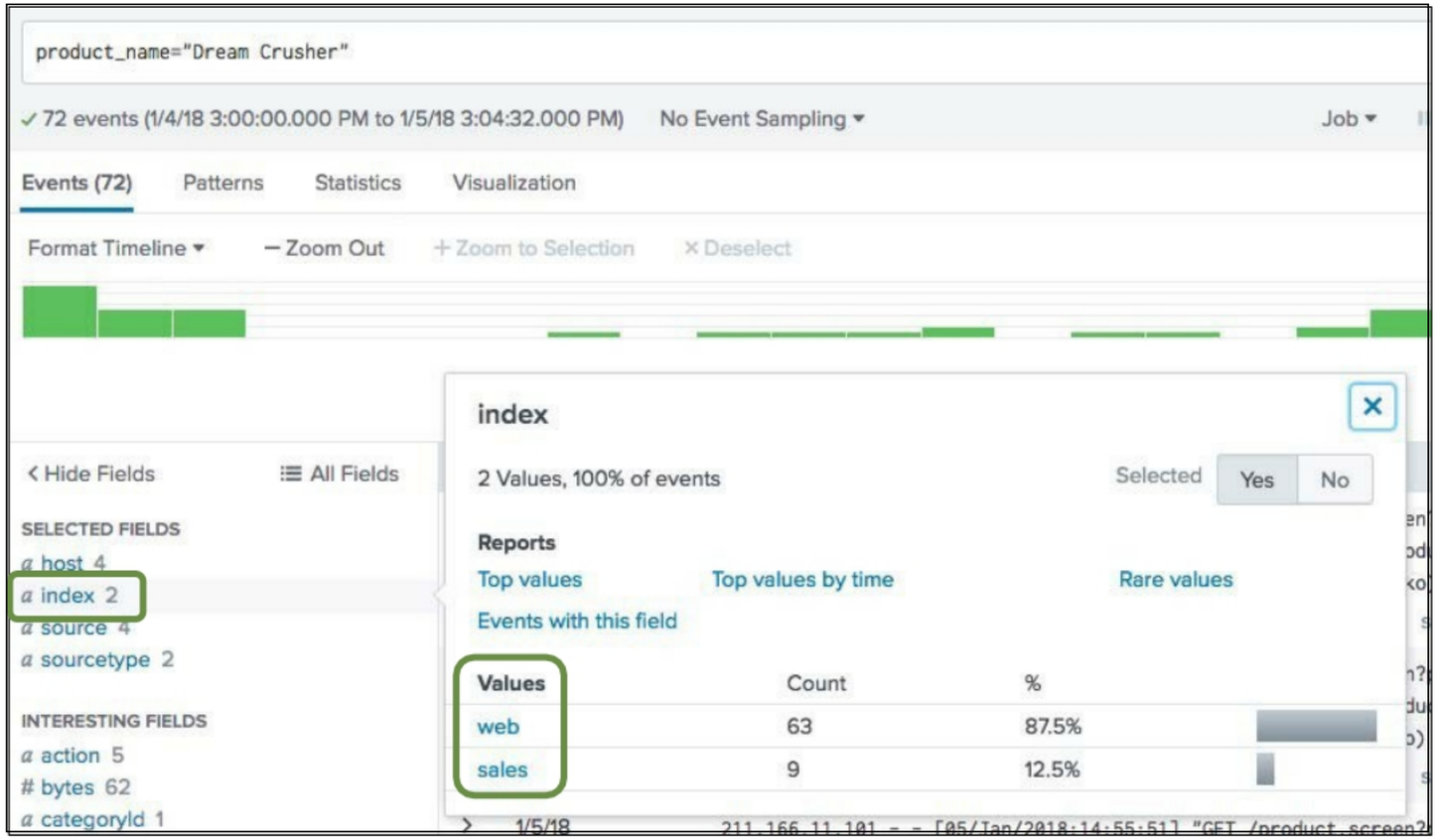
- It's possible to use a wildcard (*) in index values



- It's also possible to search *without* an index—but that's inefficient and **not** recommended

Viewing the Index Field

- The *index* always appears as a field in search results
- In the search shown here, no index was indicated in the search, so data is returned fro m two indexes: web and sales
- Remember, this practice is **not** recommended—it's always more efficient to specify one or more indexes in your search

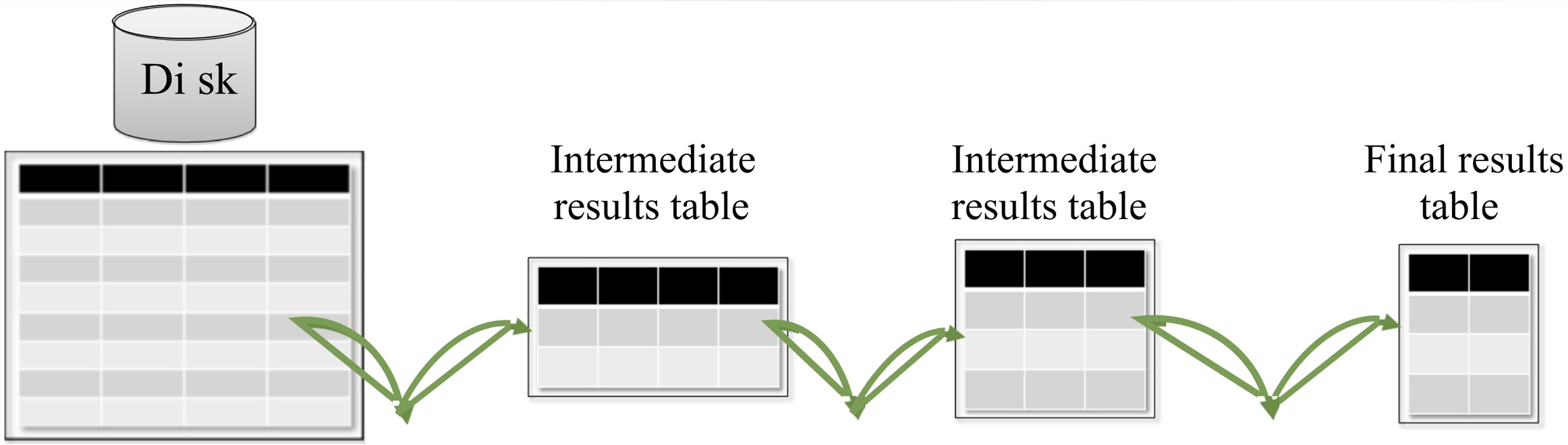


Splunk’s Search Language

Search Language Syntax Components

- Searches are made up of 5 basic components
  1. **Search terms** – what are you looking for?
     - Keywords, phrases, Booleans, etc.
  2. **Commands** – what do you want to do with the results?
     - Create a chart, compute statistics, evaluate and format, etc.
  3. **Functions** – how do you want to chart, compute, or evaluate the results?
     - Get a sum, get an average, transform the values, etc.
  4. **Arguments** – are there variables you want to apply to this function?
     - Calculate average value for a specific field, convert milliseconds to seconds, etc.
  5. **Clauses –** how do you want to group or rename the fields in the results?
     - Give a field another name or group values by or over

The Search Pipeline



index=security sourcetype=linux_secure fail*

| top user | fields – percent

Fetch events from disk that match

Summarize into table of top 10 users

Remove column showing percentage