



SECURITY
EAGLES

SPLUNK

Part 2

Introduction To Splunk

splunk >

Table of contents

Module 3: Understanding Splunk and the Data Pipeline

- An Introduction to Splunk Indexes 2
- A First Look at the Splunk Web GUI 3
- Enabling the Receiver on Splunk Enterprise 9
- Understanding Key Configuration Files 12
- Understanding and Installing Splunk Apps 16
- Shipping Windows Event Logs into Splunk 20

Module 4: Getting External Data into Splunk

- Establishing a Dedicated Index for Testing Data 24
- Ingesting Exported Windows Event Log Files 30
- Ingesting Dynamic Windows Registry Data 44
- Onboarding Generic Linux System Logs 46
- Onboarding Authentication Logs from Ubuntu UF 51
- Configuring Ingestion for Apache Web Server Access Logs 53
- Comprehensive Guide to Onboarding CSV Files 62
- Custom Data Sources and the "Great 8" Best Practices 67
- Extracting Data Fields using the EXTRACT Command 71
- Extracting Data Fields using the REPORT Command 76

An Introduction to Splunk Indexes

1. What is an Index in Splunk?

An index is like a big folder where Splunk stores all the data you send to it. It organizes the data so you can search it quickly later.

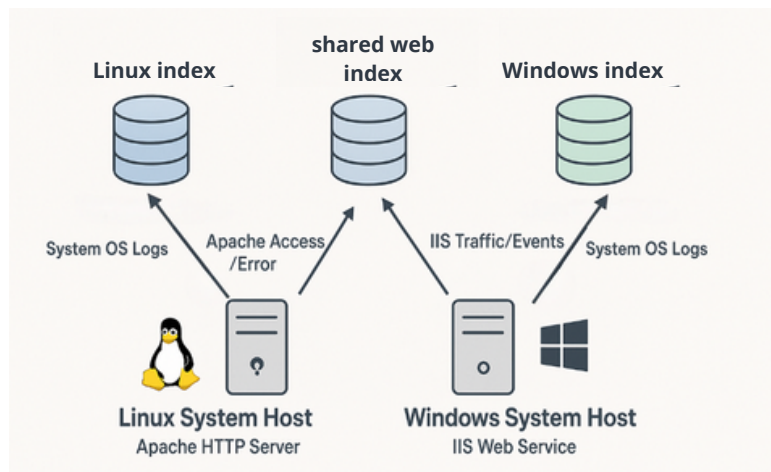
2. What does an Index do?

- **Stores Data:** It's the main place where Splunk saves your logs and information.
- **Groups Data:** You can keep similar data together (e.g., Linux logs in one index, web logs in another).
- **Controls Access:** You decide who can see which index (security).
- **Keeps Data for a Time:** You set how long Splunk keeps the data before deleting it.

3. Types of Indexes:

- **Events Index:** For normal logs and messages with time stamps.
- **Metrics Index:** For numbers like CPU usage or network speed. It's faster and uses less space for this kind of data.

4. How Indexes Work (Example):



Indexes help organize your data so searches are faster and easier.

Here's how it works:

- Linux logs → go into an index called linux_logs.
- Windows logs → go into an index called windows_logs.
- Web server logs → can go into a shared index called web_access.

This structure makes searching and analysis simple because you know exactly where to look.

A First Look at the Splunk Web GUI

The Splunk Web interface is your main workspace for searching and analyzing data. It's designed to be simple and powerful.

1. The Splunk Enterprise Home Page

When you log in to Splunk, the Home Page is your main dashboard. It's the central hub for everything you do in Splunk.



Key Features on the Home Page:

- **Apps Panel:** Access installed apps like Search & Reporting (for running queries), Audit Trail, and more.
- **Bookmarks & Dashboards:** Quickly open saved searches, dashboards, and reports.
- **Navigation Bar:**
 - **Administrator Menu:** Manage your account settings.
 - **Messages & Settings:** View system alerts and configure Splunk settings.
 - **Activity & Help:** Check recent user activity and access documentation.
- **Search Bar:** Find dashboards, reports, and saved searches easily.

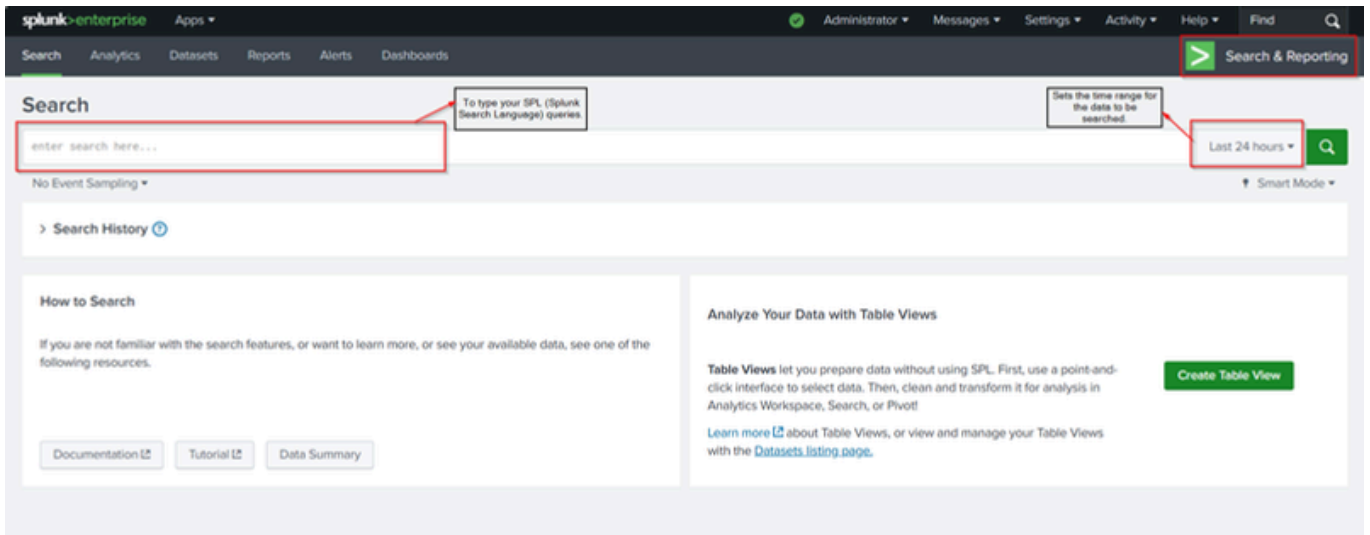
This page helps you start searches, manage data, and access dashboards all in one place.

2. Performing a Basic Search

The Search & Reporting app is where you'll spend most of your time in Splunk. It's the main tool for exploring and analyzing data.

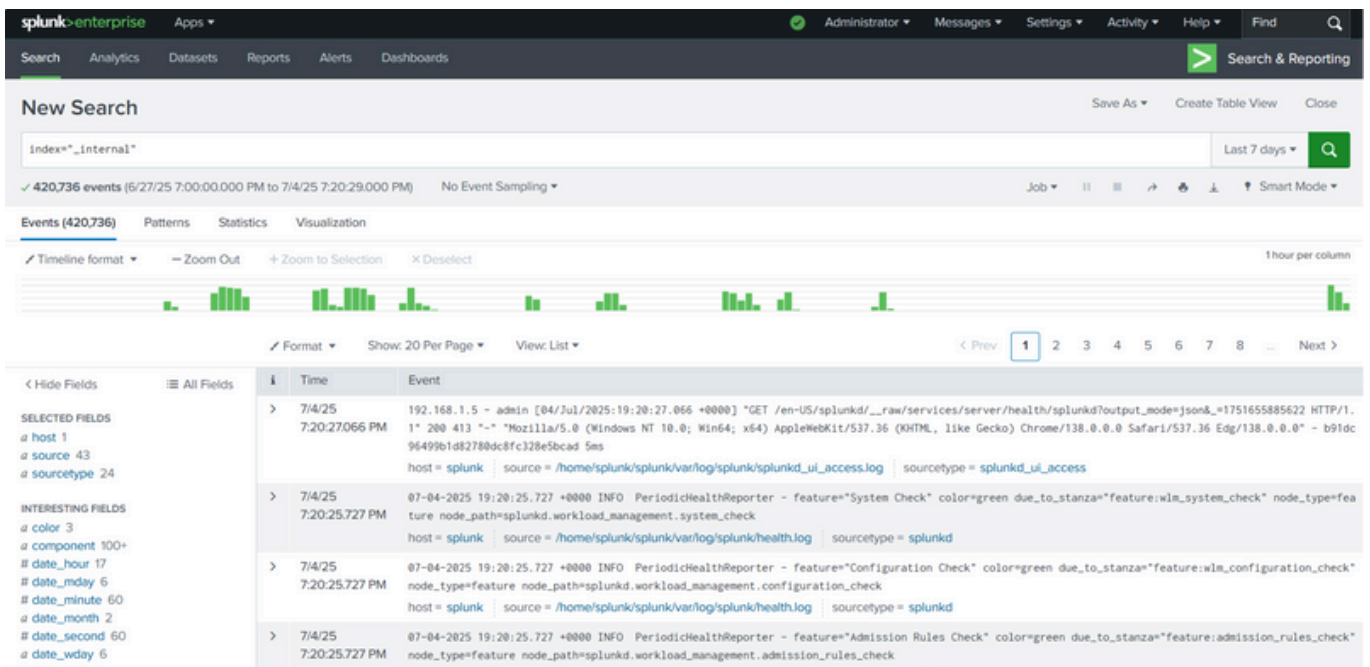
2.1 Accessing the Search Interface

- From the Home Page, click Search & Reporting (usually the first app listed).
- You'll see the Search screen with:
 - A search bar for your queries.
 - A time range picker (default: Last 24 hours).



2.2 Constructing Your Search Query

- **Search Bar:** Type your SPL (Splunk Search Language) queries here.
- **Time Range Picker:** Choose the time frame for your search.



2.4 Managing Displayed Fields

Splunk extracts numerous fields from your data. You can control which fields are shown in the event list.

The screenshot shows the Splunk interface with the event list. On the left, there are sections for 'SELECTED FIELDS' and 'INTERESTING FIELDS'. The '647 more fields' link is highlighted in a red box. The event list shows several events with their respective timestamps and details.

i	Time	Event
>	7/5/25 4:49:09.959 PM	192.168.1.5 - admin [05/Jul/2025:16:49:09.959 +0000] "GET /en-US/splunkd/_raw/services/search/jobs/1751734145.98/timeline?offset=0&count=1000&_=1751734118658 HTTP/1.1" 200 400 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/138.0.0.0 Safari/537.36 Edg/138.0.0.0" - 8e9078f6e76fd54fe88821ffc2ce5927 6ms host = splunk source = /home/splunk/splunk/var/log/splunk/splunkd_ui_access.log sourcetype = splunkd_ui_access
>	7/5/25 4:49:09.956 PM	192.168.1.5 - admin [05/Jul/2025:16:49:09.956 +0000] "GET /en-US/splunkd/_raw/servicesNS/nobody/search/search/v2/jobs/1751734145.98/summary?output_mode=json&min_freq=0&_=1751734118657 HTTP/1.1" 200 22512 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/138.0.0.0 Safari/537.36 Edg/138.0.0.0" - 8e9078f6e76fd54fe88821ffc2ce5927 93ms host = splunk source = /home/splunk/splunk/var/log/splunk/splunkd_ui_access.log sourcetype = splunkd_ui_access
>	7/5/25 4:49:09.926 PM	192.168.1.5 - admin [05/Jul/2025:16:49:09.926 +0000] "GET /en-US/splunkd/_raw/servicesNS/nobody/search/search/v2/jobs/1751734145.98/output_mode=json&_=1751734118656 HTTP/1.1" 200 3261 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/138.0.0.0 Safari/537.36 Edg/138.0.0.0" - 8e9078f6e76fd54fe88821ffc2ce5927 9ms host = splunk source = /home/splunk/splunk/var/log/splunk/splunkd_ui_access.log sourcetype = splunkd_ui_access
>	7/5/25 4:49:08.938 PM	192.168.1.5 - admin [05/Jul/2025:16:49:08.938 +0000] "GET /en-US/splunkd/_raw/services/search/jobs/1751734145.98/timeline?offset=0&count=1000&_=1751734118655 HTTP/1.1" 200 407 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/138.0.0.0 Safari/537.36 Edg/138.0.0.0" - 8e9078f6e76fd54fe88821ffc2ce5927 3ms host = splunk source = /home/splunk/splunk/var/log/splunk/splunkd_ui_access.log sourcetype = splunkd_ui_access
>	7/5/25 4:49:08.936 PM	192.168.1.5 - admin [05/Jul/2025:16:49:08.936 +0000] "GET /en-US/splunkd/_raw/servicesNS/nobody/search/search/v2/jobs/1751734145.98/summary?output_mode=json&min_freq=0&_=1751734118654 HTTP/1.1" 200 15695 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/138.0.0.0 Safari/537.36 Edg/138.0.0.0" - 8e9078f6e76fd54fe88821ffc2ce5927 31ms

- Click "647 more fields" to see all fields.

The screenshot shows the 'Select Fields' dialog. It has a search bar and a list of fields. The 'clientip' field is highlighted with a red box. The dialog also shows the number of values and event coverage for each field.

i	Field	# of Values	Event Coverage	Type
>	☒ host	1	100%	String
>	☒ source	41	100%	String
>	☒ sourcetype	22	100%	String
>	☐ active	5	1.02%	Number
>	☐ average_kbps	>100	1.75%	Number
>	☐ avg_age	>100	14.57%	Number
>	☐ blocked_count	20	3.18%	Number
>	☐ bytes	>100	18.7%	Number
>	☐ c	20	1.09%	String
>	☐ clientip	4	7.85%	String
>	☐ component	>100	89.38%	String
>	☐ ctx	70	1.09%	String

- Use Select Fields dialog to choose which fields to show.

Understanding Splunk and the Data Pipeline

Select Fields

Select All Within Filter			Deselect All		Coverage: 1% or more ▾		Filter		+ Extract New Fields	
i	✓ ▾	Field ▾	# of Values ▾		Event Coverage ▾		Type ▾			
>	☒	clientip	4		7.85%		String			
>	☒	host	1		100%		String			
>	☒	source	41		100%		String			
>	☒	sourcetype	22		100%		String			
>	☐	active	5		1.02%		Number			
>	☐	average_kbps	>100		1.75%		Number			
>	☐	avg_age	>100		14.57%		Number			
>	☐	blocked_count	20		3.18%		Number			

- Example: Selecting clientip makes IP addresses visible.

The screenshot shows the Splunk Search interface. At the top, there's a search bar with the query 'Index=internal'. Below the search bar, the results are displayed in a table. The table has two columns: 'Time' and 'Event'. The 'Event' column contains log entries from the 'splunkd' source. The 'clientip' field is highlighted in the 'Selected Fields' list on the left. The search results show three events, each with a clientip value.

- We can see clientip in the Selected Field

2.5 Filtering and Analyzing Fields

Once fields are displayed, you can further refine your search or explore field values.

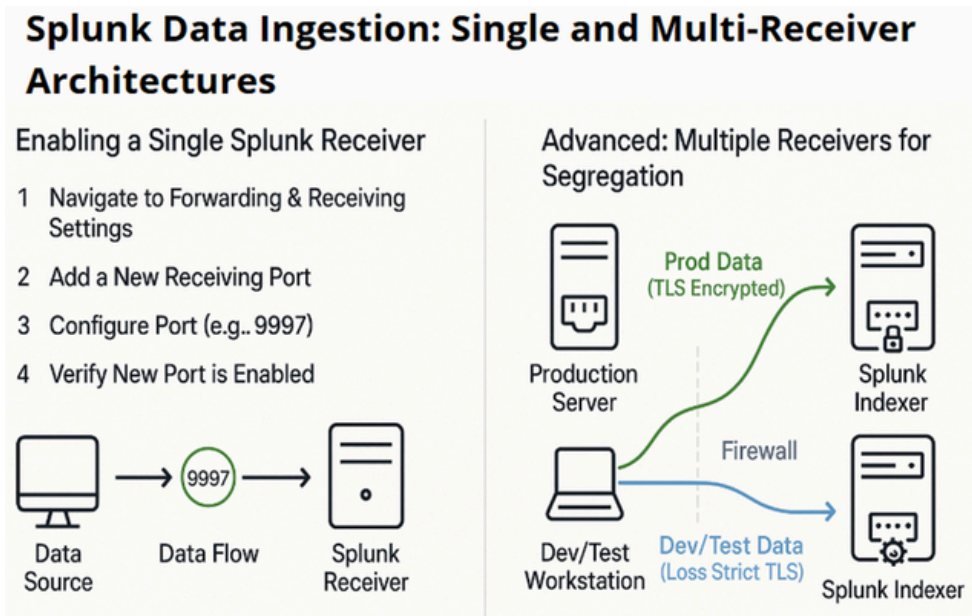
This screenshot is identical to the one above, showing the Splunk Search interface with the search query 'Index=internal'. The 'clientip' field is highlighted in the 'Selected Fields' list, and the search results table displays three log events with their corresponding clientip values.

- Click a field (e.g., clientip) to see:
 - Summary: Unique values and coverage.
 - Value Distribution: Counts and percentages.

Enabling the Receiver on Splunk Enterprise

Splunk uses receivers to accept data from forwarders. A receiver is simply a port that listens for incoming data.

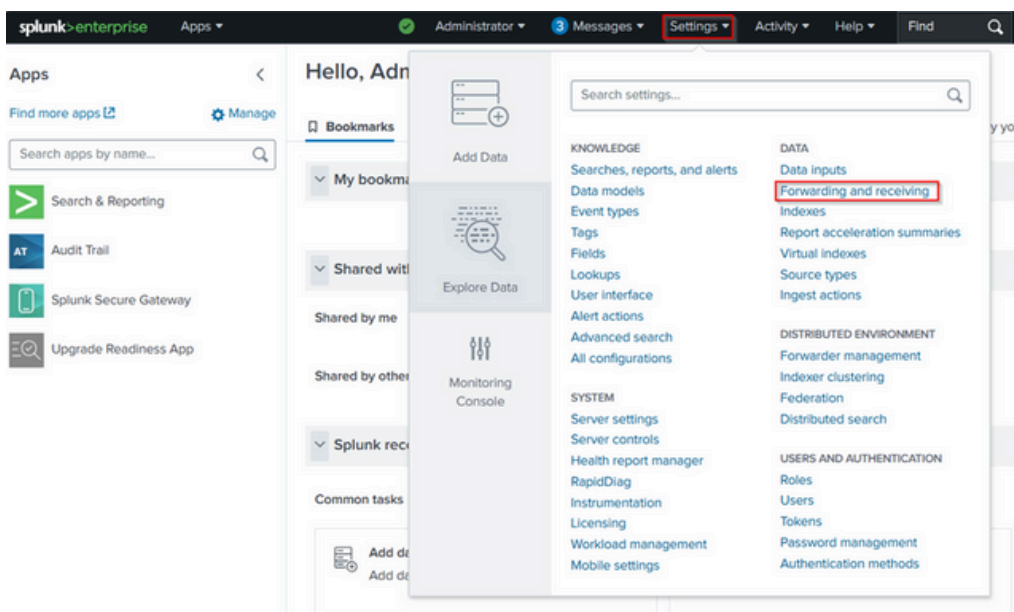
Most setups only need one receiver, but you can configure multiple for advanced environments.



Step-by-Step Guide

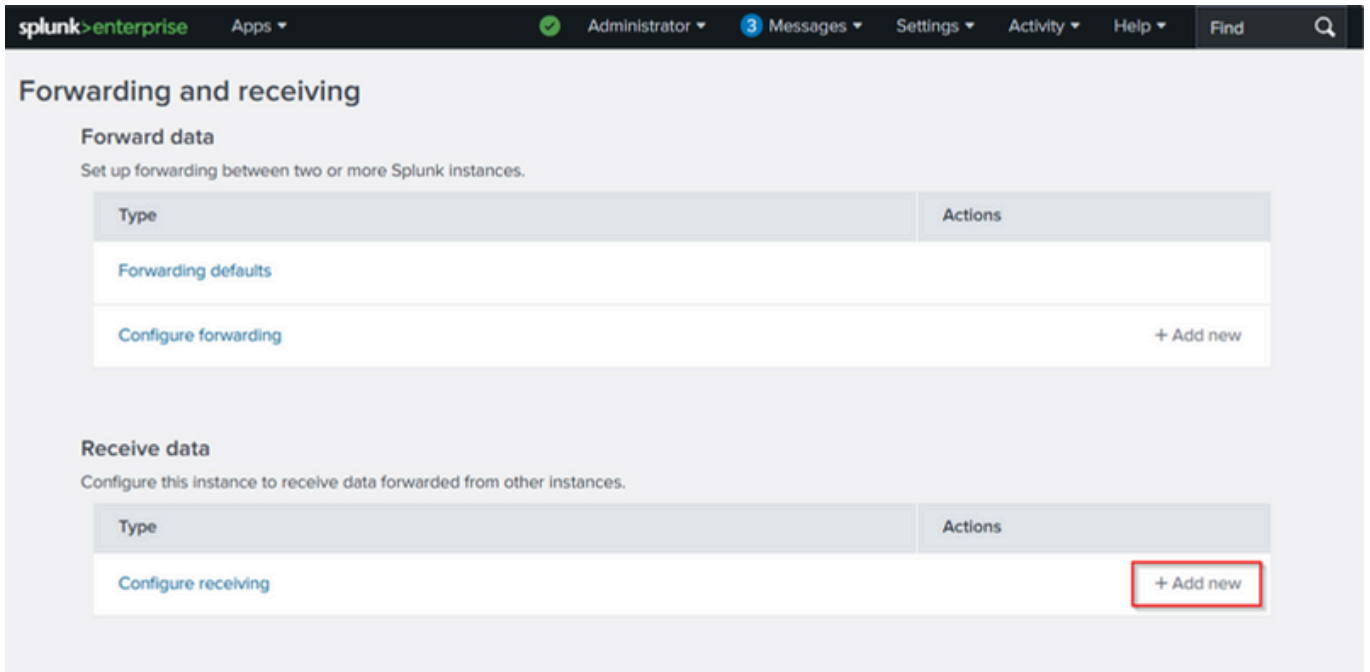
1. Navigate to Settings:

- From the Splunk Home page, click **Settings** → under **Data**, select **Forwarding and Receiving**.



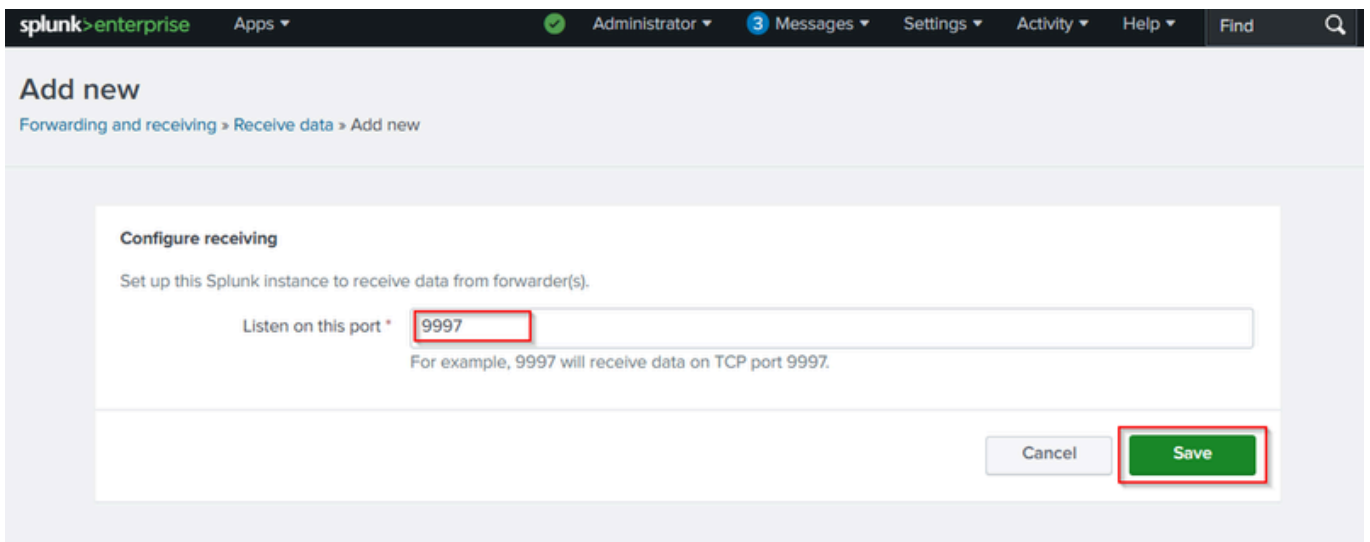
2. Add a New Receiving Port:

- On the **Forwarding and Receiving** page, find **Receive data** and click **+ Add new** next to **Configure receiving**.



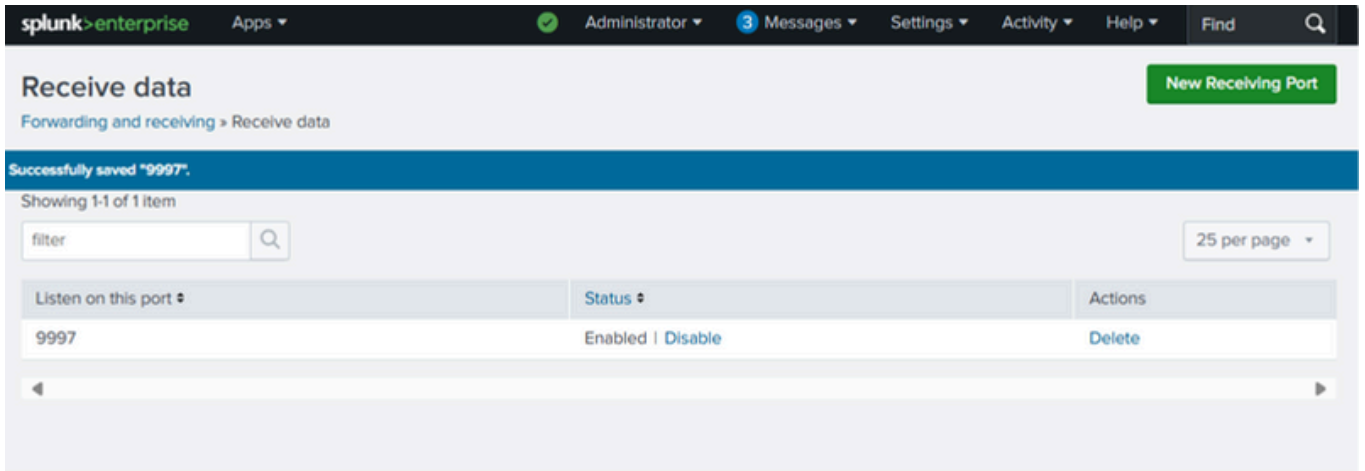
3. Configure the Receiving Port:

- Enter the port number in **Listen on this port** (commonly **9997**).
- Click **Save**.



4. Verify:

- You'll see a confirmation message like: "Successfully saved '9997'."
- The port will appear with **Status: Enabled**.



Understanding Key Configuration Files

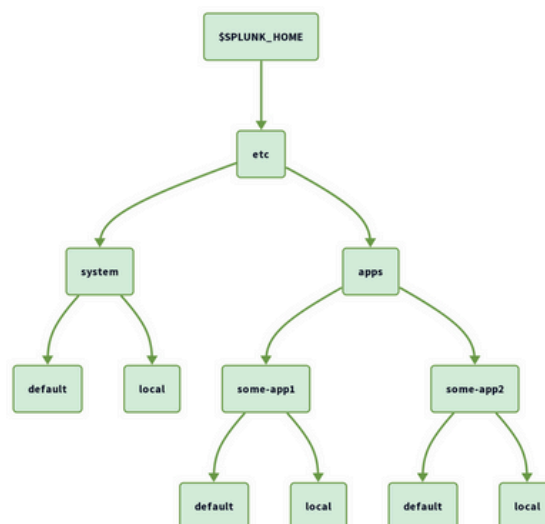
Splunk uses configuration files to control how it works. These files let you manage things like data inputs, parsing rules, server settings, and the web interface. Knowing how they are organized and which file takes priority is essential for administration.

Splunk Configuration File Overview

- Splunk settings are stored in **.conf files**, each for a specific purpose:
 - inputs.conf → Data inputs
 - props.conf → Parsing rules
 - transforms.conf → Data transformations
 - server.conf → Server settings
 - web.conf → Web interface
 - indexes.conf → Index settings
- These files are spread across the **\$SPLUNK_HOME/etc** directory, not in one place.
- Each file uses **stanzas** (sections in square brackets []) to organize settings.

Navigating Splunk Configuration Directories

- Main directory: \$SPLUNK_HOME/etc
- Inside, you'll find:
 - system/ → Global settings
 - apps/ → App-specific settings
- Each has:
 - default/ → Original settings (don't edit here)
 - local/ → Your custom changes (safe to edit)



For instance, within a Splunk installation (e.g., at /home/splunk/splunk), navigating to \$SPLUNK_HOME/etc/system/default reveals a collection of default configuration files:

```
splunk@splunk:~$ sudo su
[sudo] password for splunk:
root@splunk:/home/splunk# ls
splunk  splunk-9.4.2-e9664af3d956-linux-amd64.tgz  splunk-9.4.3-237ebbd22314-linux-amd64.tgz
root@splunk:/home/splunk# cd ./splunk
root@splunk:/home/splunk/splunk# ls
bin          include      LICENSE.txt  quarantined_files  splunk-9.4.2-e9664af3d956-linux-amd64-manifest
copyright.txt lib          openssl     README-splunk.txt  swidtag
etc          license-eula.txt  opt         share              var
root@splunk:/home/splunk/splunk# cd etc
root@splunk:/home/splunk/splunk/etc# ls
anonymizer      init.d          log-debug.cfg      myinstall          splunk-launch.conf.default
apps            instance.cfg    login-info.cfg      openldap           splunk.version
auth            licenses        log-searchprocess.cfg  packages          system
copyright.txt   log-btool.cfg   log-tlsproxy.cfg    passwd            users
datetime.xml    log-btool-debug.cfg  log-utility.cfg     prettyprint.xsl
deployment-apps log.cfg          manager-apps        shcluster
disabled-apps   log-cmdline.cfg   master-apps         splunk-enttrial.lic
findlogs.ini    log-cmdline-debug.cfg  modules            splunk-launch.conf
root@splunk:/home/splunk/splunk/etc# cd system/
root@splunk:/home/splunk/splunk/etc/system# ls
bin  default  local  lookups  metadata  README  static
root@splunk:/home/splunk/splunk/etc/system# cd default/
root@splunk:/home/splunk/splunk/etc/system/default# ls
agent_management.conf  default-mode.conf  ipc_broker.conf  savedsearches.conf  ui-tour.conf
alert_actions.conf     distsearch.conf    limits.conf       searchbnf.conf       viewstates.conf
app.conf               eventdiscoverer.conf  literals.conf     segmenters.conf      visualizations.conf
audit.conf             event_renderers.conf  livetail.conf     serverclass.conf     web.conf
authentication.conf     federated.conf        messages.conf     server.conf          web-features.conf
authorize.conf          field_filters.conf    metric_alerts.conf  source-classifier.conf  workflow_actions.conf
collections.conf        fields.conf           metric_rollups.conf  sourcetypes.conf     workload_policy.conf
commands.conf          global-banner.conf    multikv.conf        telemetry.conf        workload_pools.conf
conf.conf              health.conf           outputs.conf         times.conf            workload_rules.conf
data                   indexes.conf          procmon-filters.conf  transactiontypes.conf  transforms.conf
datamodels.conf        inputs.conf           props.conf           ui-prefs.conf
datatypesbnf.conf      restmap.conf
```

Understanding Configuration File Precedence

When Splunk starts, it merges all files with the same name. If there's a conflict, this is the priority (highest to lowest):

1. \$SPLUNK_HOME/etc/system/local
2. \$SPLUNK_HOME/etc/apps/<app>/local
3. \$SPLUNK_HOME/etc/apps/<app>/default
4. \$SPLUNK_HOME/etc/system/default

Tip: Always make changes in local/ directories, not default.

Example: web.conf

The web.conf file controls **Splunk's web interface settings**.

```
root@splunk:/home/splunk/splunk/etc/system/default# nano web.conf
```

```
GNU nano 6.2 web.conf
# Version 9.4.2
# DO NOT EDIT THIS FILE!
# Changes to default files will be lost on update and are difficult to
# manage and support.
#
# Please make any changes to system defaults by overriding them in
# apps or $SPLUNK_HOME/etc/system/local
# (See "Configuration file precedence" in the web documentation).
#
# To override a specific setting, copy the name of the stanza and
# setting to the file where you wish to override it.
#
# This file contains possible attributes and values you can use to configure Splunk's web interface.
#
[default]

[settings]

# enable/disable the appserver
startwebserver = 1

# port number tag is missing or 0 the server will NOT start an http listener
# this is the port used for both SSL and non-SSL (we only have 1 port now).
httpport = 8000

# this determines whether to start SplunkWeb in http or https.
enableSplunkWebSSL = false

# location of splunkd; don't include http[s]:// in this anymore.
mgmtHostPort = 127.0.0.1:8089

# list of ports to start python application servers on (although usually
# one port is enough)
#
# In the past a special value of "0" could be passed here to disable
# the modern UI appserver infrastructure, but that is no longer supported.
appServerPorts = 8065

# default timeout, in seconds, when communicating with splunkd
splunkdConnectionTimeout = 30

# enable/disable custom netloc when using http client
enableSplunkWebClientNetloc = False

# SSL certificate files.
privKeyPath = $SPLUNK_HOME/etc/auth/splunkweb/privkey.pem
serverCert = $SPLUNK_HOME/etc/auth/splunkweb/cert.pem
```

What this means:

- startwebserver = true → Splunk Web will start.
- httpport = 8000 → The web interface runs on port 8000.
- enableSplunkWebSSL = false → SSL is disabled (not secure).

Different locations might have different values:

Location	Settings
\$SPLUNK_HOME/etc/system/local/web.conf	enableSplunkWebSSL = true
\$SPLUNK_HOME/etc/apps/some-app/local/web.conf	(none)
\$SPLUNK_HOME/etc/apps/some-app/default/web.conf	httpport = 443
\$SPLUNK_HOME/etc/system/default/web.conf	startwebserver = true httpport = 8000 enableSplunkWebSSL = false

Final Effective Settings

- startwebserver = true (from system/default, not overridden)
- httpport = 443 (from app/default, overrides default)
- enableSplunkWebSSL = true (from system/local, highest priority)

Why this matters:

- Always make changes in local/ directories, not default.
- This ensures your settings persist after upgrades and follow best practices.

Understanding and Installing Splunk Apps

Splunk apps are packages that extend Splunk's functionality. They can add dashboards, reports, integrations, and more.

1. What Are Splunk Apps?

- **Self-contained packages** (e.g., .tgz, .tar.gz) that enhance Splunk Enterprise or Universal Forwarder.
- **What they do:**
 - **Enhance Data Processing:** Add field extractions and parsing rules.
 - **Provide Visualizations:** Dashboards, charts, and reports.
 - **Enable Integrations:** Connect Splunk with external tools.
 - **Streamline Operations:** Pre-built alerts and dashboards.

App Directory Structure

- bin/ → Scripts and custom code.
- default/ → Default configs (don't edit here).
- local/ → Your custom changes (safe to edit).
- lookups/ → Lookup tables for enrichment.
- metadata/ → Permissions and metadata.
- static/ → Icons and images.

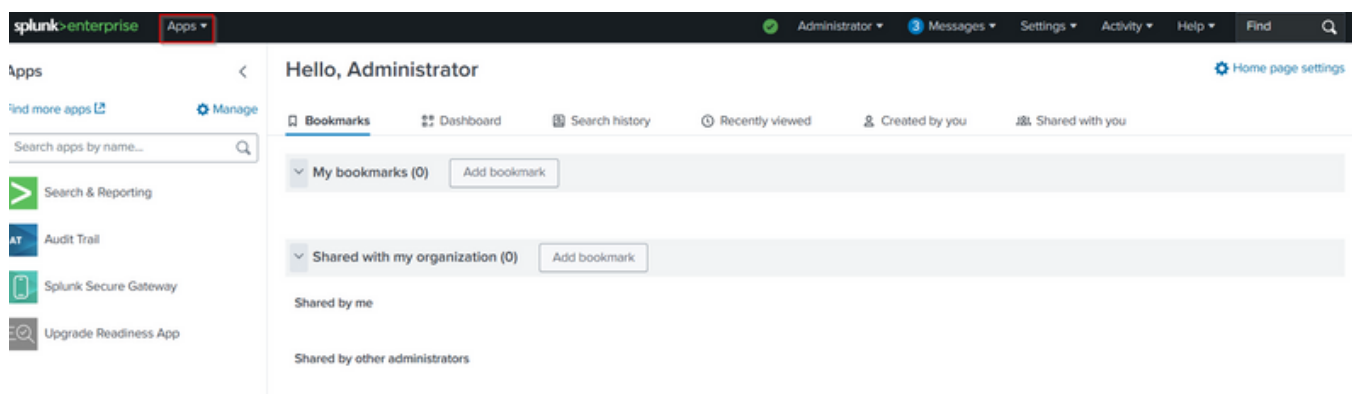
2. Installing Splunk Apps

Two main methods:

2.1 Installation via Splunk Web Interface

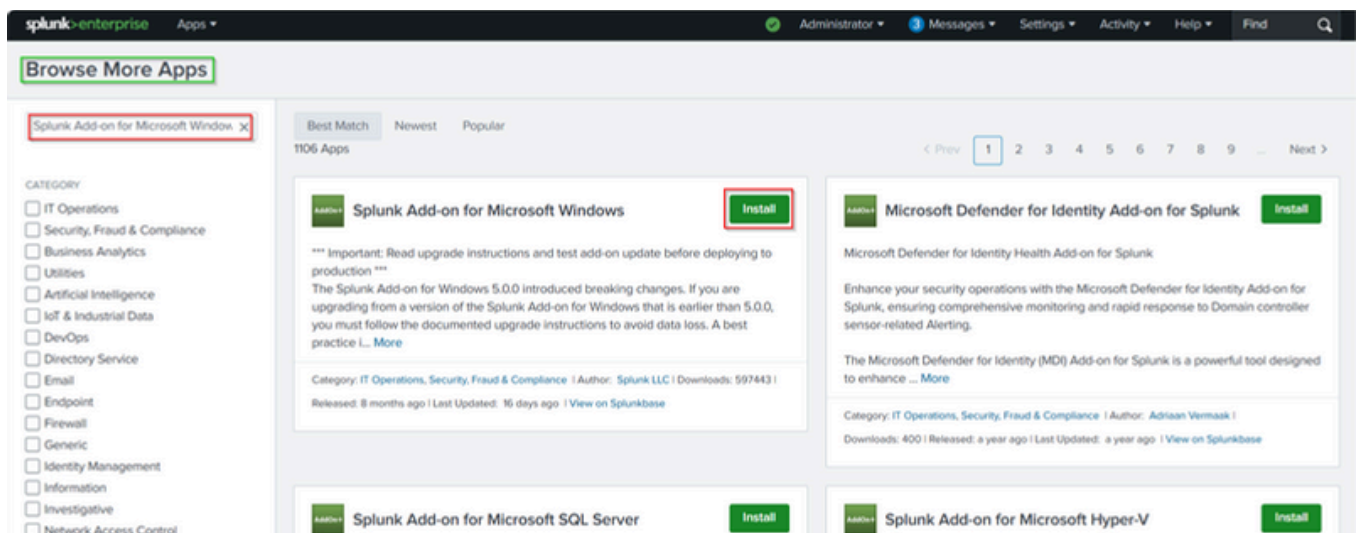
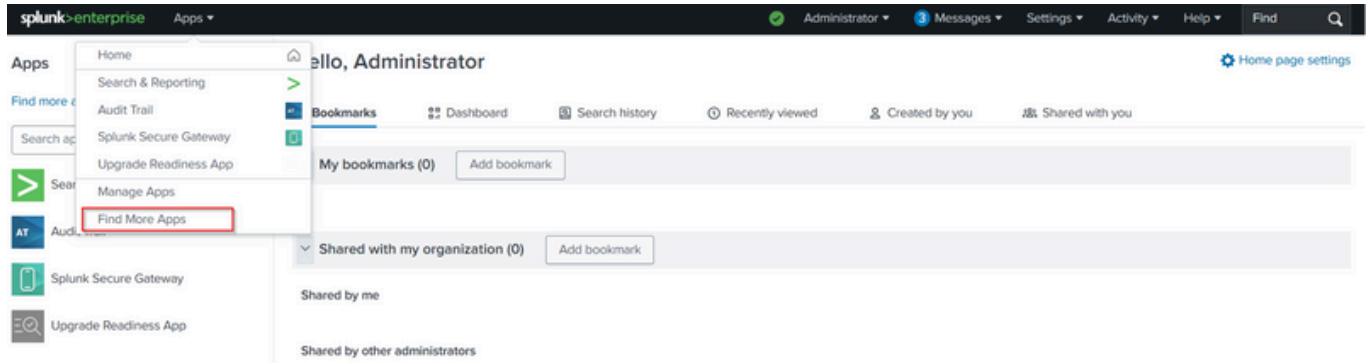
Steps:

1. **Access App Management:** Go to **Apps** from the Splunk homepage.



2. Choose Installation Method:

- Browse More Apps (Splunkbase): Search and install directly.
- Install from File: Upload a .tgz file.



Login and Install

Enter your Splunk.com username and password to download the app.

[Forgot your password?](#)

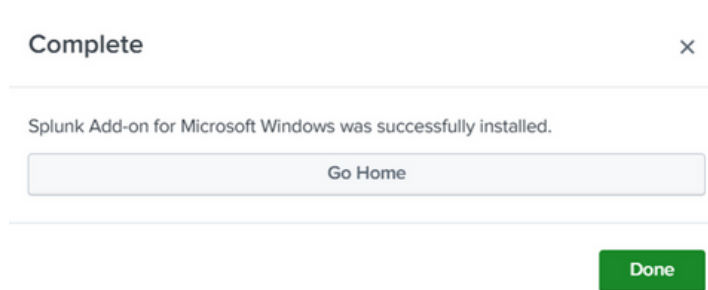
The app, and any related dependency that will be installed, may be provided by Splunk and/or a third party and your right to use these app(s) is in accordance with the applicable license(s) provided by Splunk and/or the third-party licensor. Splunk is not responsible for any third-party app (developed by you or a third party) and does not provide any warranty or support. Installation of a third-party app can introduce security risks. By clicking "Agree" below, you acknowledge and accept such risks. If you have any questions, complaints or claims with respect to an app, please contact the applicable licensor directly whose contact information can be found on the Splunkbase download page.

Splunk Add-on for Microsoft Windows is governed by the following license: [sgt](#)

I have read the terms and conditions of the license(s) and agree to be bound by them. I also agree to Splunk's [Website Terms of Use](#).

3. Upload and Install: Splunk processes the package.

4. Completion: Confirmation message appears. Some apps may require a restart.



2.2 Verify Installation

- On the server:

```
splunk@splunk:~$ cd ./splunk/etc/apps
```

- Use `ls -al` to list installed apps.

```
splunk@splunk:~/splunk/etc/apps$ ls -al
total 124
drwxr-xr-x 31 splunk splunk 4096 Jul  6 20:29 .
drwxr-xr-x 18 splunk splunk 4096 Jul  6 19:42 ..
drwxr-xr-x  7 splunk splunk 4096 Mar 18 22:42 alert_logevent
drwxr-xr-x  7 splunk splunk 4096 Mar 18 22:42 alert_webhook
drwxr-xr-x  4 splunk splunk 4096 Mar 18 22:42 appsbrowser
drwxr-xr-x  6 splunk splunk 4096 Mar 18 22:42 audit_trail
drwxr-xr-x  5 splunk splunk 4096 Mar 18 22:42 introspection_generator_addon
drwxr-xr-x  4 splunk splunk 4096 Mar 18 22:42 journald_input
drwxr-xr-x  7 splunk splunk 4096 Mar 18 22:42 launcher
drwxr-xr-x  5 splunk splunk 4096 Jun 28 13:49 learned
drwxr-xr-x  4 splunk splunk 4096 Mar 18 22:42 legacy
drwxr-xr-x  9 splunk splunk 4096 Jun 28 20:12 python_upgrade_readiness_app
drwxr-xr-x  6 splunk splunk 4096 Mar 18 22:42 sample_app
drwxr-xr-x  8 splunk splunk 4096 Mar 18 22:42 search
drwxr-xr-x  6 splunk splunk 4096 Jun 28 19:41 splunk_archiver
drwxr-xr-x  9 splunk splunk 4096 Jun 28 13:49 splunk-dashboard-studio
drwxr-xr-x  4 splunk splunk 4096 Mar 18 22:42 SplunkDeploymentServerConfig
drwxr-xr-x  4 splunk splunk 4096 Mar 18 22:42 SplunkForwarder
drwxr-xr-x  7 splunk splunk 4096 Mar 18 22:45 splunk_gdi
drwxr-xr-x  3 splunk splunk 4096 Mar 18 22:42 splunk_httpinput
drwxr-xr-x  9 splunk splunk 4096 Jun 28 13:49 splunk_instrumentation
drwxr-xr-x  4 splunk splunk 4096 Mar 18 22:42 splunk_internal_metrics
drwxr-xr-x  4 splunk splunk 4096 Mar 18 22:42 SplunkLightForwarder
drwxr-xr-x  8 splunk splunk 4096 Mar 18 22:44 splunk_metrics_workspace
drwxr-xr-x  8 splunk splunk 4096 Jun 28 13:49 splunk_monitoring_console
drwxr-xr-x 10 splunk splunk 4096 Jul  5 16:43 splunk_rapid_diag
drwxr-xr-x  8 splunk splunk 4096 Jul 19 20:24 splunk-rolling-upgrade
drwxr-xr-x 10 splunk splunk 4096 Feb 18 16:04 splunk_secure_gateway
drwxr-xr-x 10 splunk splunk 4096 Jul  6 20:29 Splunk_TA_windows
drwxr-xr-x  5 splunk splunk 4096 Nov 12 20:24 splunk-visual-exporter
drwxr-xr-x  4 splunk splunk 4096 Mar 18 22:42 user-prefs
```

- Explore default/ for original configs and local/ for custom changes.

```
splunk@splunk:~/splunk/etc/apps$ cd Splunk_TA_windows/default/
splunk@splunk:~/splunk/etc/apps/Splunk_TA_windows/default$ ls -al
total 252
drwxr-xr-x  3 splunk splunk  4096 Jul  6 20:29 .
drwxr-xr-x 10 splunk splunk  4096 Jul  6 20:29 ..
-rw-----  1 splunk splunk   558 Jul  6 20:29 app.conf
drwxr-xr-x  3 splunk splunk  4096 Jul  6 20:29 data
-rw-r--r--  1 splunk splunk 26329 Jul  6 20:29 eventtypes.conf
-rw-r--r--  1 splunk splunk 20145 Jul  6 20:29 inputs.conf
-rw-r--r--  1 splunk splunk  2407 Jul  6 20:29 macros.conf
-rw-r--r--  1 splunk splunk 119677 Jul  6 20:29 props.conf
-rw-r--r--  1 splunk splunk  12264 Jul  6 20:29 tags.conf
-rw-r--r--  1 splunk splunk  40173 Jul  6 20:29 transforms.conf
-rw-r--r--  1 splunk splunk   4450 Jul  6 20:29 wmi.conf
-rw-r--r--  1 splunk splunk   1589 Jul  6 20:29 workflow_actions.conf
```

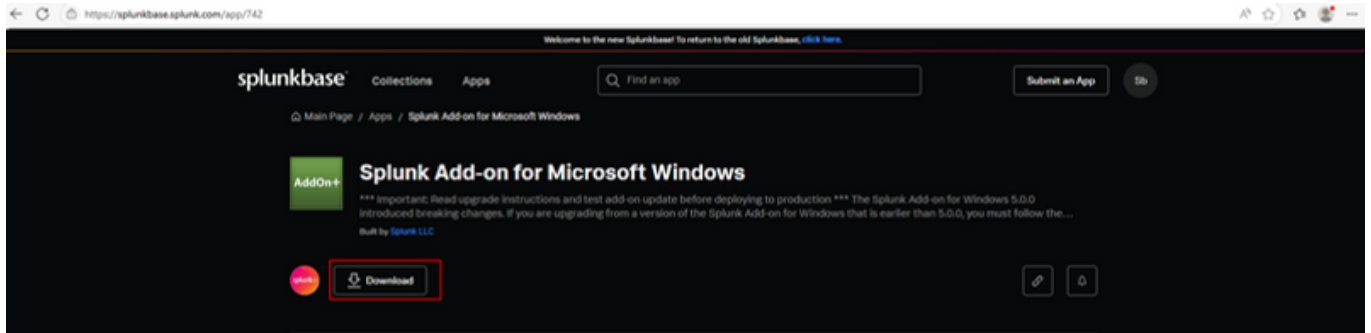
Best Practice: Always edit in local/, never in default/.

Shipping Windows Event Logs into Splunk

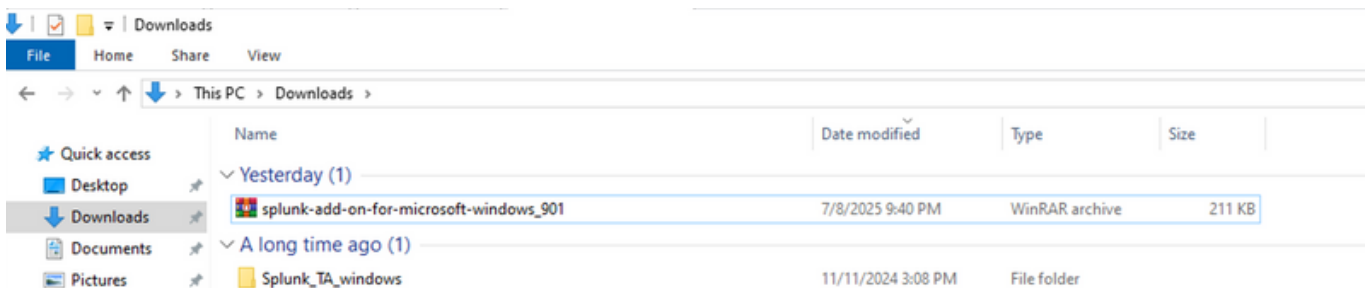
To collect Windows event logs and send them to Splunk, you use a Splunk Universal Forwarder with the Splunk Add-on for Microsoft Windows.

Step 1: Download the Add-on

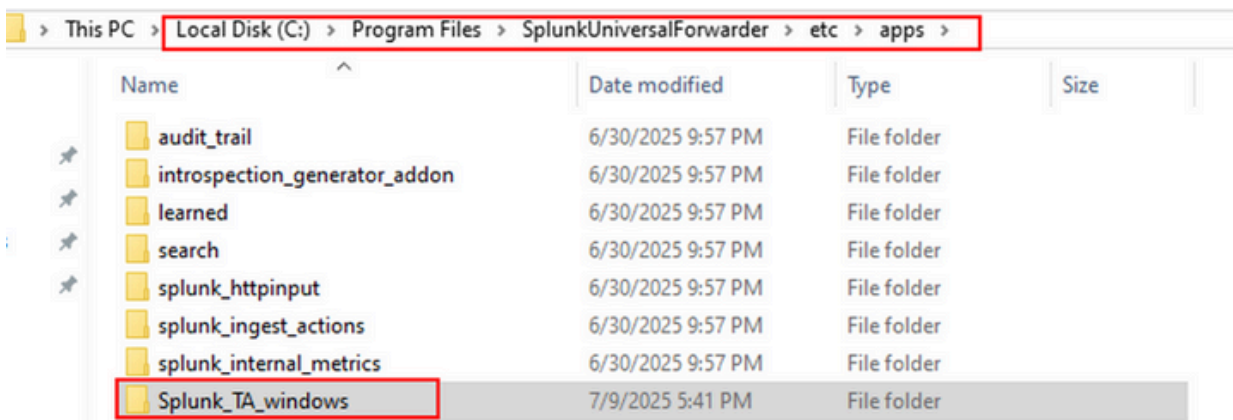
- Go to Splunkbase (<https://splunkbase.splunk.com/app/742>) and download **Splunk Add-on for Microsoft Windows**.



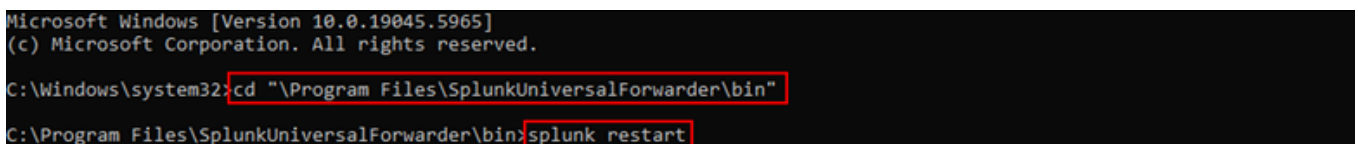
- Extract it to:



- Make sure the folder name is:



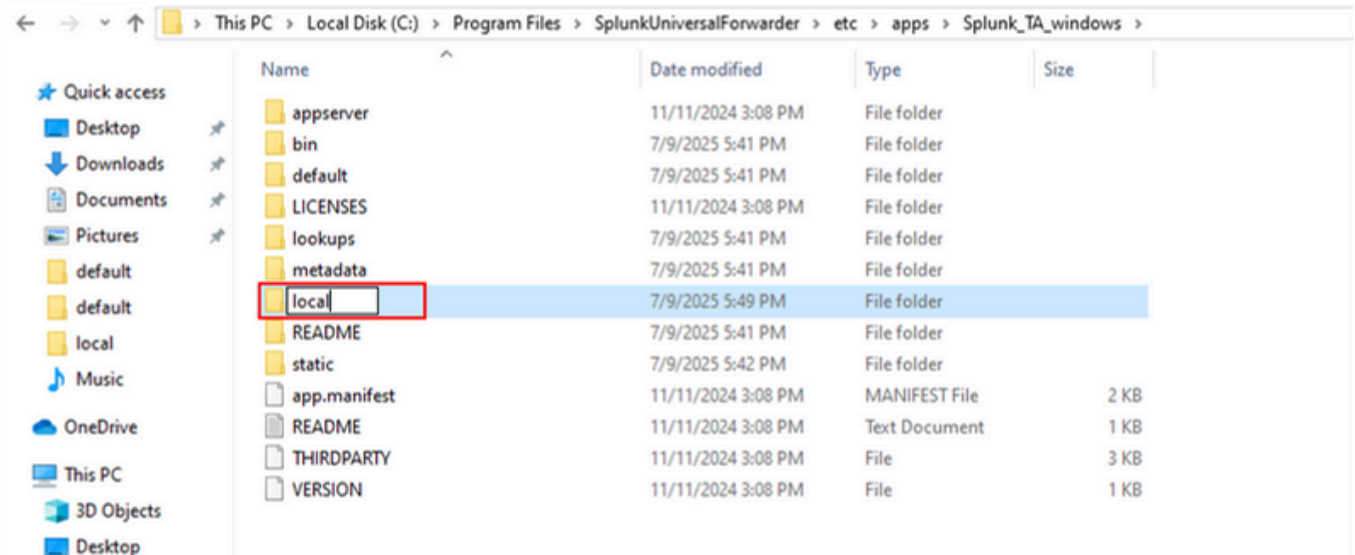
- Restart



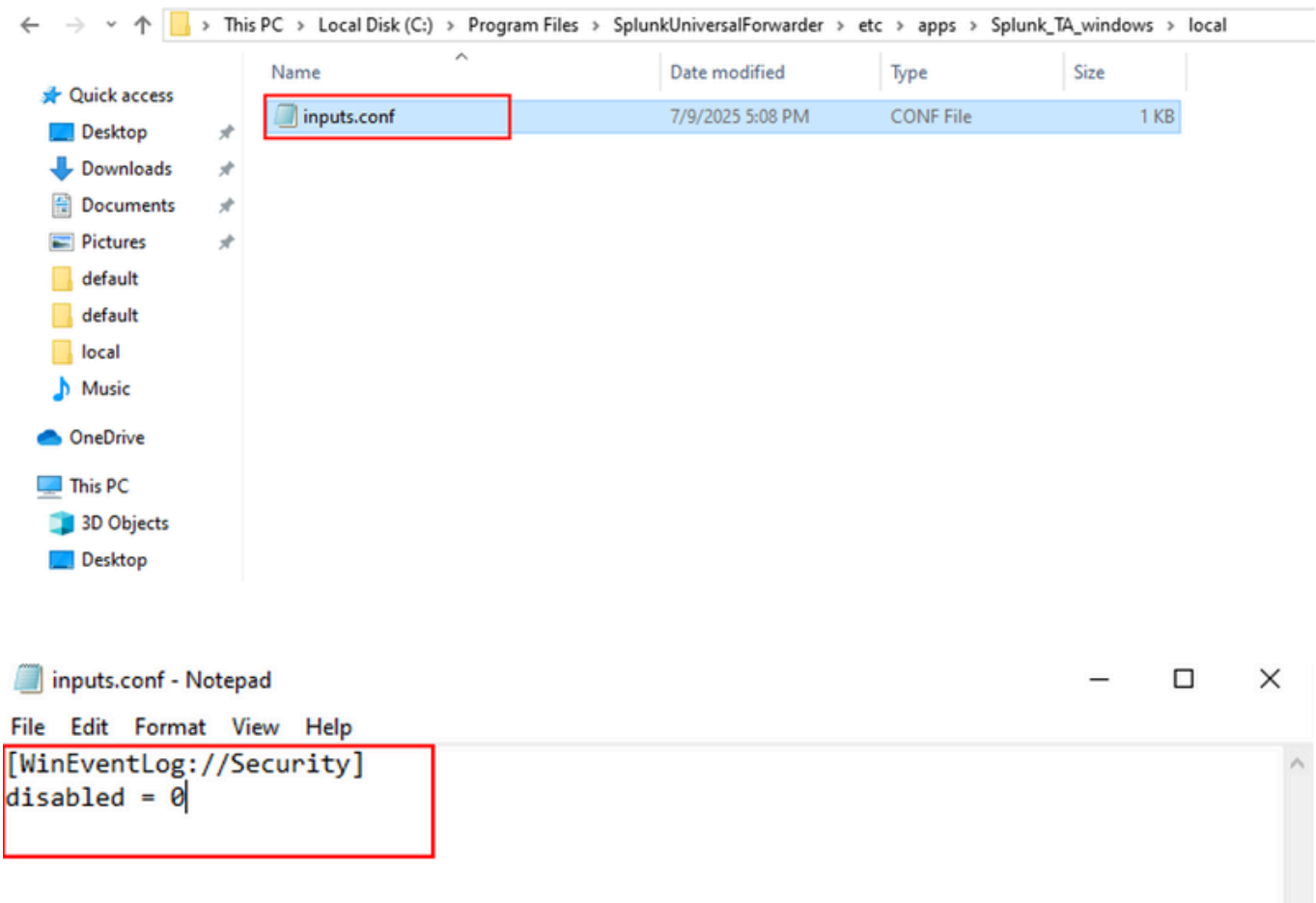
Step 2: Configure inputs.conf

- Path:

C:\Program Files\SplunkUniversalForwarder\etc\apps\Splunk_TA_windows\local\



- Open inputs.conf and add:

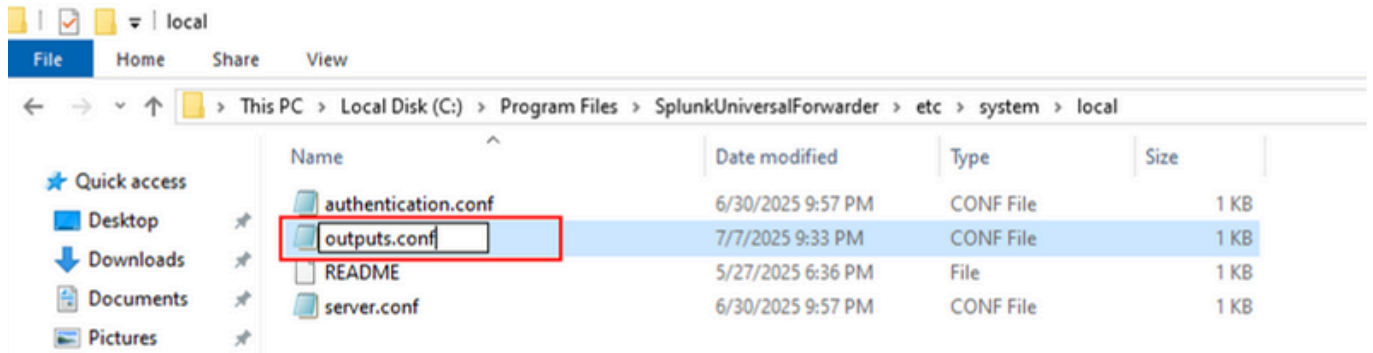


- This enables collection of Windows Security event logs.

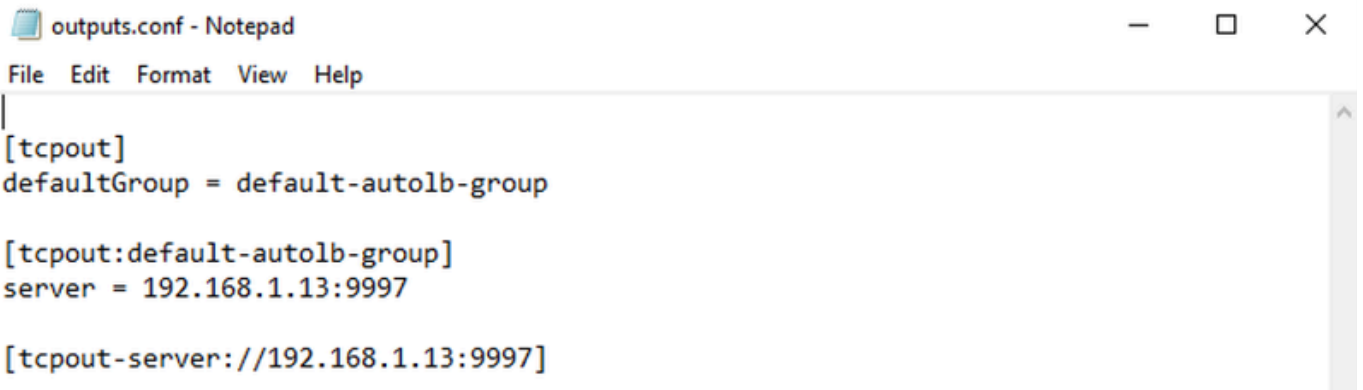
Step 3: Configure outputs.conf

- Path:

C:\Program Files\SplunkUniversalForwarder\etc\system\local\



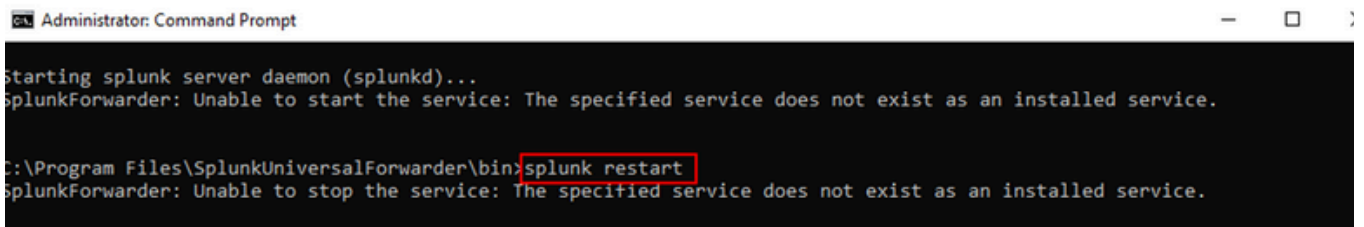
- Open outputs.conf and add:



- Replace 192.168.1.13 with your Splunk indexer IP and 9997 with the receiving port.

Step 4: Restart the Forwarder

- Open Command Prompt (Admin):



Step 5: Verify in Splunk

- In Splunk Web, search:

The screenshot shows the Splunk Web interface. At the top, there's a navigation bar with 'splunk-enterprise' and 'Apps'. Below it, a search bar contains 'index=main'. The search results show 9,643 events from 7/9/25 4:00:00 PM to 7/9/25 4:31:00 PM. The visualization shows a timeline with two events at 8 AM on Thursday, July 10, 2025. The search results table shows two events:

Time	Event
7/9/25 4:29:45.000 PM	<Event xmlns="http://schemas.microsoft.com/win/2004/8/6/events/event"><System><Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54839255-5478-4994-a5ba-be3b8328384d}" /><EventID>4672</EventID><Version></Version><Level></Level><Task>12544</Task><Opcode></Opcode><Keywords>0x8020000000000000</Keywords><TimeCreated SystemTime="2025-07-10T16:29:45.0666932Z" /><EventRecordID>9643</EventRecordID><Correlation ActivityID="{72627239-f1b8-4001-a3f2-6272b0f1b081}" /><Execution ProcessID="652" ThreadID="728" /><Channel>Security</Channel><Computer>DESKTOP-SR75NGF</Computer><Security /></System><EventData><Data Name="SubjectUserSid">NT SERVICE\SplunkForwarder</Data><Data Name="SubjectUserName">SplunkForwarder</Data><Data Name="SubjectDomainName">NT SERVICE</Data><Data Name="SubjectLogonId">0x1284846</Data><Data Name="PrivilegeList">SeSecurityPrivilege SeBackupPrivilege SeImpersonatePrivilege</Data></EventData></Event>
7/9/25 4:29:45.000 PM	<Event xmlns="http://schemas.microsoft.com/win/2004/8/6/events/event"><System><Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54839255-5478-4994-a5ba-be3b8328384d}" /><EventID>4624</EventID><Version></Version><Level></Level><Task>12544</Task><Opcode></Opcode><Keywords>0x8020000000000000</Keywords><TimeCreated SystemTime="2025-07-10T16:29:45.0666932Z" /><EventRecordID>9642</EventRecordID><Correlation ActivityID="{72627239-f1b8-4001-a3f2-6272b0f1b081}" /><Execution ProcessID="652" ThreadID="728" /><Channel>Security</Channel><Computer>DESKTOP-SR75NGF</Computer><Security /></System><EventData><Data Name="SubjectUserSid">AUTHORITY\NT SYSTEM</Data><Data Name="SubjectUserName">DESKTOP-SR75NGF</Data><Data Name="SubjectDomainName">WORKGROUP</Data><Data Name="SubjectLogonId">0x3e7</Data><Data Name="TargetUserSid">NT SERVICE\SplunkForwarder</Data><Data Name="TargetUserName">SplunkForwarder</Data><Data Name="TargetDomainName">NT SERVICE</Data><Data Name="TargetLogonId">0x1284846</Data><Data Name="Logon Type">5</Data><Data Name="LogonProcessName">Advapi</Data><Data Name="AuthenticationPackage">Negotiate</Data><Data Name="WorkstationName"></Data><Data Name="LogonGuid">{00000000-0000-0000-0000-00000000}</Data><Data Name="TransmittedServices"></Data><Data Name="LpPackage"></Data><Data Name="KeyLength">0</Data><Data Name="ProcessId">0x278</Data><Data Name="ProcessName">C:\Windows\System32

If events appear, your setup is successful.

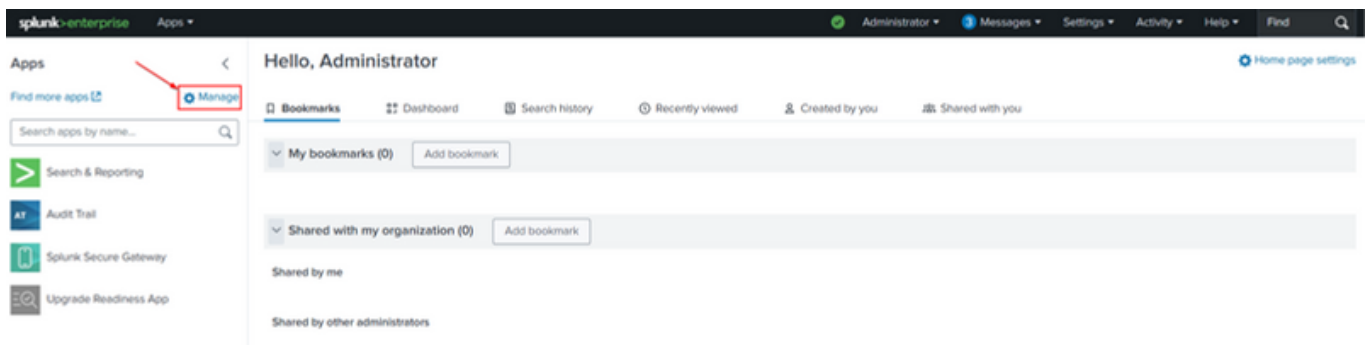
Establishing a Dedicated Index for Testing Data

When working with custom data or lab environments, it's best to create a dedicated index. This keeps test data separate from production and makes cleanup easier.

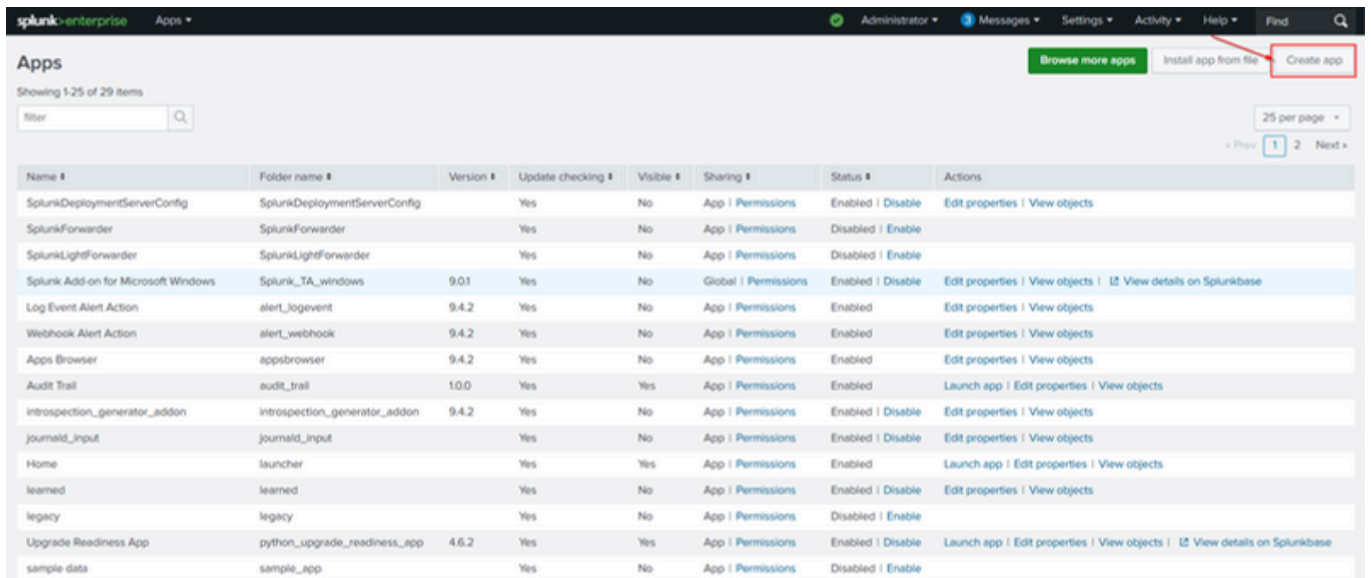
1. Creating a New Splunk App

To better organize your custom configurations, it's recommended to create a dedicated Splunk app.

1. Go to **Manage**: In Splunk Web (browser interface), click Manage.



2. Click **"Create App"**.



3. Fill the form like this:

- Folder name: indexes
- Version: 1.0.0
- Visible: No (because it has no views/pages)
- Author: Thomas Feller
- Description: Our own indexes to test
- Template: barebones
- Upload asset: Leave it empty

Getting External Data into Splunk

Name
Give your app a friendly name for display in Splunk Web.

Folder name *
This name maps to the app's directory in \$SPLUNK_HOME/etc/apps/.

Version
App version.

Visible ☒ No ☐ Yes
Only apps with views should be made visible.

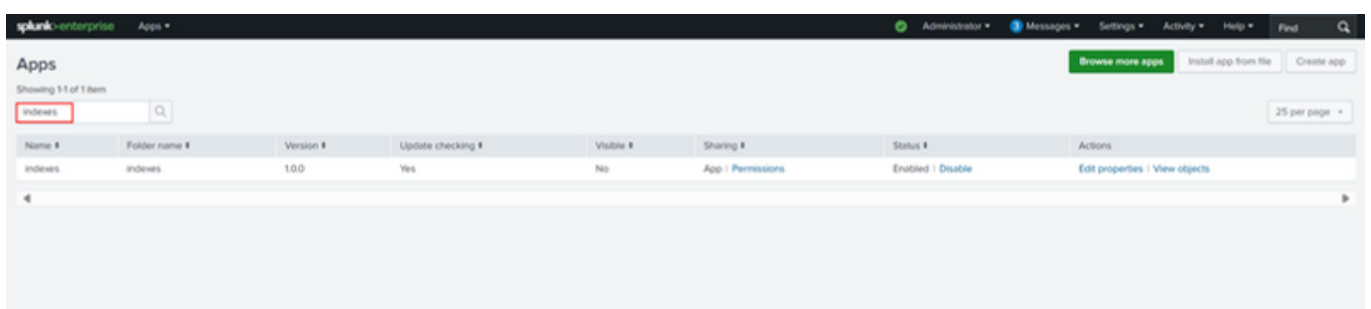
Author
Name of the app's owner.

Description
Enter a description for your app.

Template
These templates contain example views and searches.

Upload asset No file chosen
Can be any html, js, or other file to add to your app.

4. Click **Save** to finish creating the app.



2. Check the App Configuration

After creating the app, you can check its settings using the command line (terminal).

Steps:

1. **Go to the app folder** on your server.

```
splunk@splunk:~/splunk/etc/apps$ ls -al | grep indexes
drwx--x--- 6 splunk splunk 4096 Jul 8 19:42 indexes
splunk@splunk:~/splunk/etc/apps$ cd indexes/
splunk@splunk:~/splunk/etc/apps/indexes$ ls -al
total 24
drwx--x--- 6 splunk splunk 4096 Jul 8 19:42 .
drwxr-xr-x 33 splunk splunk 4096 Jul 8 19:42 ..
drwx--x--- 2 splunk splunk 4096 Jul 8 19:42 bin
drwx--x--- 3 splunk splunk 4096 Jul 8 19:42 default
drwx----- 2 splunk splunk 4096 Jul 8 19:42 local
drwx--x--- 2 splunk splunk 4096 Jul 8 19:42 metadata
splunk@splunk:~/splunk/etc/apps/indexes$ cd default/
splunk@splunk:~/splunk/etc/apps/indexes/default$ ls -al
total 16
drwx--x--- 3 splunk splunk 4096 Jul 8 19:42 .
drwx--x--- 6 splunk splunk 4096 Jul 8 19:42 ..
-rw----- 1 splunk splunk 193 Jul 8 19:42 app.conf
drwx--x--- 3 splunk splunk 4096 Jul 8 19:42 data
```

2. **Open app.conf file**, this file shows the details you entered when creating the app.

```
splunk@splunk:~/splunk/etc/apps/indexes/default$ cat app.conf
#
# Splunk app configuration file
#

[install]
is_configured = 0

[ui]
is_visible = 0
label = indexes

[launcher]
author = Salma Benmina
description = Our own indexes to test
version = 1.0.0
```

3. Create a New Index (Using Configuration File)

An index is where Splunk stores data.

Steps:

1. Inside your app folder, go to the default directory.
2. Create or edit a file called indexes.conf.

```
splunk@splunk:~/splunk/etc/apps/indexes/default$ nano indexes.conf
```

3. Add this configuration to define a new index called test:

```
GNU nano 6.2                               indexes.conf *
[test]
homePath = $SPLUNK_DB/$_index_name/db
coldPath = $SPLUNK_DB/$_index_name/colddb
thawedPath = $SPLUNK_DB/test/thaweddb
```

- **[test]**: Defines a new index named test.
- **homePath**: Where hot/warm data is stored.
- **coldPath**: Where cold data is stored.
- **thawedPath**: Where restored (thawed) data is stored.

4. Restart Splunk

To apply the new index settings, you need to restart Splunk.

```
splunk@splunk:~/splunk/bin$ ./splunk restart
Stopping splunkd...
Shutting down. Please wait, as this may take a few minutes.
==== AUTHENTICATING FOR org.freedesktop.systemd1.manage-units ====
Authentication is required to stop 'Splunkd.service'.
Authenticating as: splunk
Password:
==== AUTHENTICATION COMPLETE ====

Stopping splunk helpers...

Done.
splunkd.pid doesn't exist...

Splunk> Winning the War on Error

Checking prerequisites...
  Checking http port [8000]: open
  Checking mgmt port [8089]: open
  Checking appserver port [127.0.0.1:8065]: open
  Checking kvstore port [8191]: open
  Checking configuration... Done.
  Checking critical directories... Done
  Checking indexes...
    Validated: _audit _configtracker _dsappevent _dsclient _dsphonehome _internal _introspection _metrics _m
etrics_rollup _telemetry _thefishbucket history main summary test
    Done
  Checking filesystem compatibility... Done
  Checking conf files for problems...
```

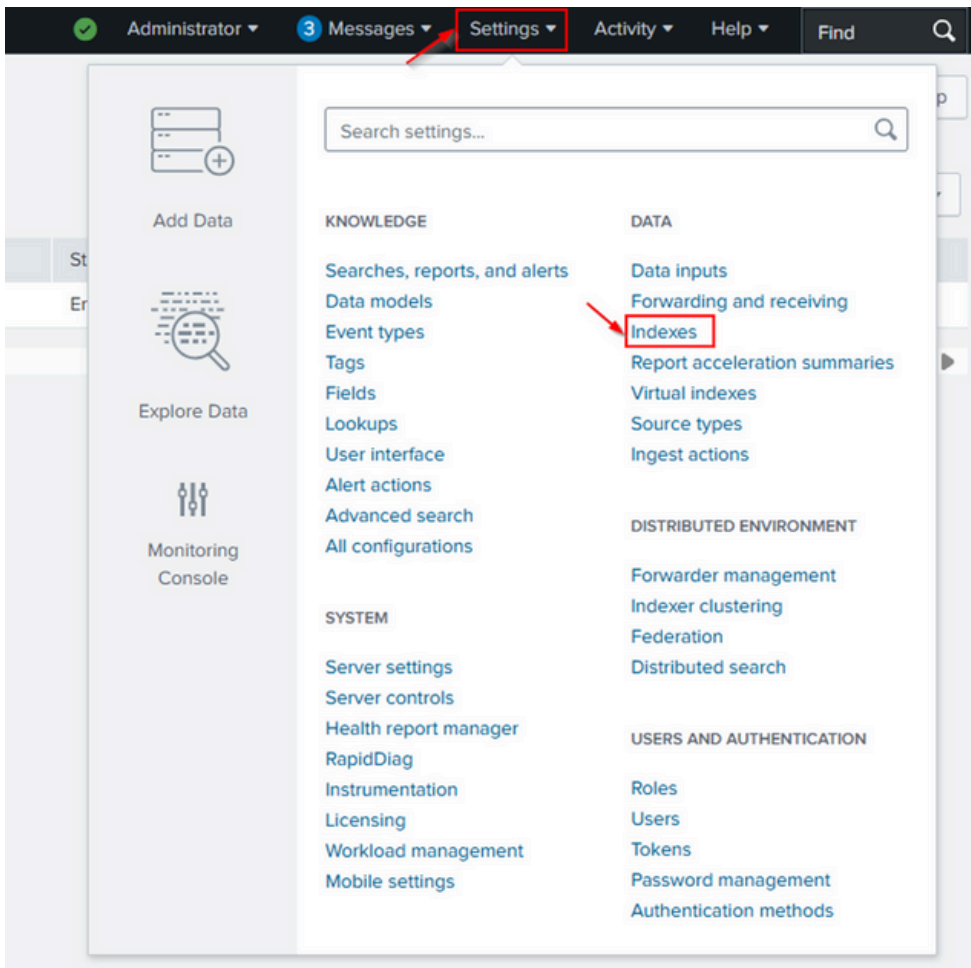
- You'll be asked for your password to confirm the restart.

5. Check the New Index in Splunk Web

After restart:

1. Go to **Settings > Indexes** in Splunk Web.

Getting External Data into Splunk



2. You should see the new index called test listed there.

Indexes

A repository for data in Splunk Enterprise. Indexes reside in flat files on the Splunk Enterprise instance known as the index. [Learn more](#)

16 indexes 20 per page

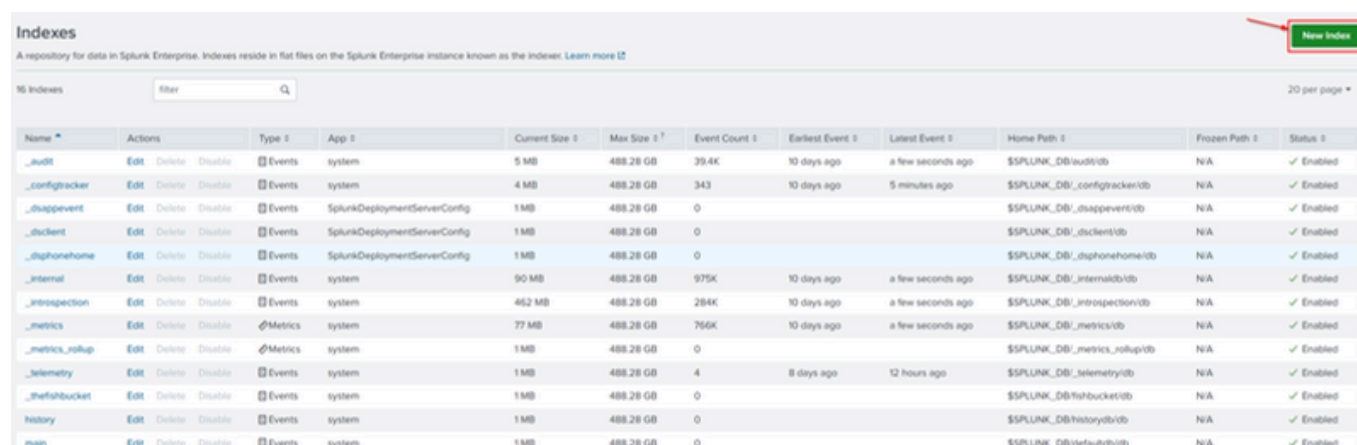
Name	Actions	Type	App	Current Size	Max Size	Event Count	Earliest Event	Latest Event	Home Path	Frozen Path	Status
audit	Edit Delete Disable	Events	system	5 MB	488.28 GB	39.4K	10 days ago	a few seconds ago	\$SPLUNK_DB/auditdb	N/A	✓ Enabled
configtracker	Edit Delete Disable	Events	system	4 MB	488.28 GB	343	10 days ago	5 minutes ago	\$SPLUNK_DB/configtrackerdb	N/A	✓ Enabled
disappevent	Edit Delete Disable	Events	SplunkDeploymentServerConfig	1 MB	488.28 GB	0			\$SPLUNK_DB/disappeventdb	N/A	✓ Enabled
dsckent	Edit Delete Disable	Events	SplunkDeploymentServerConfig	1 MB	488.28 GB	0			\$SPLUNK_DB/dsckentdb	N/A	✓ Enabled
dtphonehome	Edit Delete Disable	Events	SplunkDeploymentServerConfig	1 MB	488.28 GB	0			\$SPLUNK_DB/dtphonehome	N/A	✓ Enabled
internal	Edit Delete Disable	Events	system	90 MB	488.28 GB	975K	10 days ago	a few seconds ago	\$SPLUNK_DB/internaldb	N/A	✓ Enabled
introspection	Edit Delete Disable	Events	system	462 MB	488.28 GB	284K	10 days ago	a few seconds ago	\$SPLUNK_DB/introspectiondb	N/A	✓ Enabled
metrics	Edit Delete Disable	Metrics	system	77 MB	488.28 GB	766K	10 days ago	a few seconds ago	\$SPLUNK_DB/metricsdb	N/A	✓ Enabled
metrics_rollup	Edit Delete Disable	Metrics	system	1 MB	488.28 GB	0			\$SPLUNK_DB/metrics_rollupdb	N/A	✓ Enabled
telemetry	Edit Delete Disable	Events	system	1 MB	488.28 GB	4	8 days ago	12 hours ago	\$SPLUNK_DB/telemetrydb	N/A	✓ Enabled
thefishbucket	Edit Delete Disable	Events	system	1 MB	488.28 GB	0			\$SPLUNK_DB/thefishbucketdb	N/A	✓ Enabled
history	Edit Delete Disable	Events	system	1 MB	488.28 GB	0			\$SPLUNK_DB/historydb	N/A	✓ Enabled
main	Edit Delete Disable	Events	system	1 MB	488.28 GB	0			\$SPLUNK_DB/defaultdb	N/A	✓ Enabled
splunklogger	Edit Delete Disable	Events	system	0 B	488.28 GB	0			\$SPLUNK_DB/splunkloggerdb	N/A	✗ Disabled
summary	Edit Delete Disable	Events	system	1 MB	488.28 GB	0			\$SPLUNK_DB/summarydb	N/A	✓ Enabled
test	Edit Delete Disable	Events	indexes	1 MB	488.28 GB	0			\$SPLUNK_DB/test_index_namedb	N/A	✓ Enabled

6. Create a New Index Manually (Alternative Method)

You can also create an index directly from the Splunk Web interface — no need to edit files.

Steps:

1. Go to **Settings > Indexes** in Splunk Web.
2. Click **"New Index"**.



Indexes
A repository for data in Splunk Enterprise. Indexes reside in flat files on the Splunk Enterprise instance known as the index. [Learn more](#)

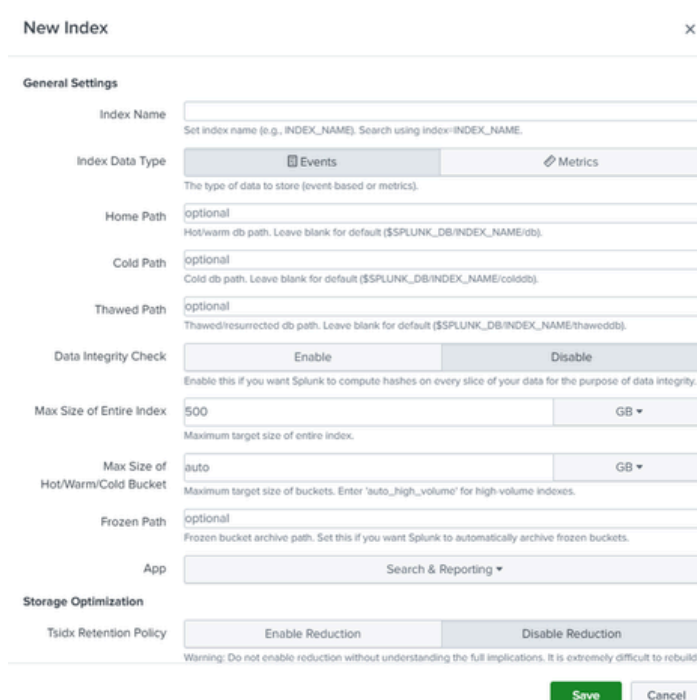
16 Indexes 20 per page

Name	Actions	Type	App	Current Size	Max Size	Event Count	Earliest Event	Latest Event	Home Path	Frozen Path	Status
audit	Edit Delete Disable	Events	system	5 MB	488.28 GB	39.4K	10 days ago	a few seconds ago	\$SPLUNK_DB/audit/db	N/A	✓ Enabled
_configtrack	Edit Delete Disable	Events	system	4 MB	488.28 GB	343	10 days ago	5 minutes ago	\$SPLUNK_DB/_configtrack/db	N/A	✓ Enabled
_disappevent	Edit Delete Disable	Events	SplunkDeploymentServerConfig	1 MB	488.28 GB	0			\$SPLUNK_DB/_disappevent/db	N/A	✓ Enabled
_disclient	Edit Delete Disable	Events	SplunkDeploymentServerConfig	1 MB	488.28 GB	0			\$SPLUNK_DB/_disclient/db	N/A	✓ Enabled
_disphonehome	Edit Delete Disable	Events	SplunkDeploymentServerConfig	1 MB	488.28 GB	0			\$SPLUNK_DB/_disphonehome/db	N/A	✓ Enabled
_internal	Edit Delete Disable	Events	system	90 MB	488.28 GB	975K	10 days ago	a few seconds ago	\$SPLUNK_DB/_internal/db	N/A	✓ Enabled
_introspection	Edit Delete Disable	Events	system	462 MB	488.28 GB	284K	10 days ago	a few seconds ago	\$SPLUNK_DB/_introspection/db	N/A	✓ Enabled
_metrics	Edit Delete Disable	Metrics	system	77 MB	488.28 GB	766K	10 days ago	a few seconds ago	\$SPLUNK_DB/_metrics/db	N/A	✓ Enabled
_metrics_rollup	Edit Delete Disable	Metrics	system	1 MB	488.28 GB	0			\$SPLUNK_DB/_metrics_rollup/db	N/A	✓ Enabled
_telemetry	Edit Delete Disable	Events	system	1 MB	488.28 GB	4	8 days ago	12 hours ago	\$SPLUNK_DB/_telemetry/db	N/A	✓ Enabled
_fishbucket	Edit Delete Disable	Events	system	1 MB	488.28 GB	0			\$SPLUNK_DB/fishbucket/db	N/A	✓ Enabled
history	Edit Delete Disable	Events	system	1 MB	488.28 GB	0			\$SPLUNK_DB/history/db	N/A	✓ Enabled
main	Edit Delete Disable	Events	system	1 MB	488.28 GB	0			\$SPLUNK_DB/default/db	N/A	✓ Enabled

3. Fill in the form:

- **Index Name:** Type the name you want (e.g., test)
- **Index Data Type:** Choose Events or Metrics
- **Home Path / Cold Path / Thawed Path:** You can leave them empty to use default paths, or set custom ones.
- **Data Integrity Check:** Turn it on or off depending on your needs.

4. Click **Save** to create the index.



New Index

General Settings

Index Name:

Set index name (e.g., INDEX_NAME). Search using index=INDEX_NAME.

Index Data Type: ☒ Events ☐ Metrics

The type of data to store (event based or metrics).

Home Path:

optional
Hot/warm db path. Leave blank for default (\$SPLUNK_DB/INDEX_NAME/db).

Cold Path:

optional
Cold db path. Leave blank for default (\$SPLUNK_DB/INDEX_NAME/colddb).

Thawed Path:

optional
Thawed/resurrected db path. Leave blank for default (\$SPLUNK_DB/INDEX_NAME/thaweddb).

Data Integrity Check: ☒ Enable ☐ Disable

Enable this if you want Splunk to compute hashes on every slice of your data for the purpose of data integrity.

Max Size of Entire Index: GB

Maximum target size of entire index.

Max Size of Hot/Warm/Cold Bucket: GB

Maximum target size of buckets. Enter 'auto_high_volume' for high-volume indexes.

Frozen Path:

Frozen bucket archive path. Set this if you want Splunk to automatically archive frozen buckets.

App:

Storage Optimization

Tsxid Retention Policy: ☒ Enable Reduction ☐ Disable Reduction

Warning: Do not enable reduction without understanding the full implications. It is extremely difficult to rebuild

Save **Cancel**

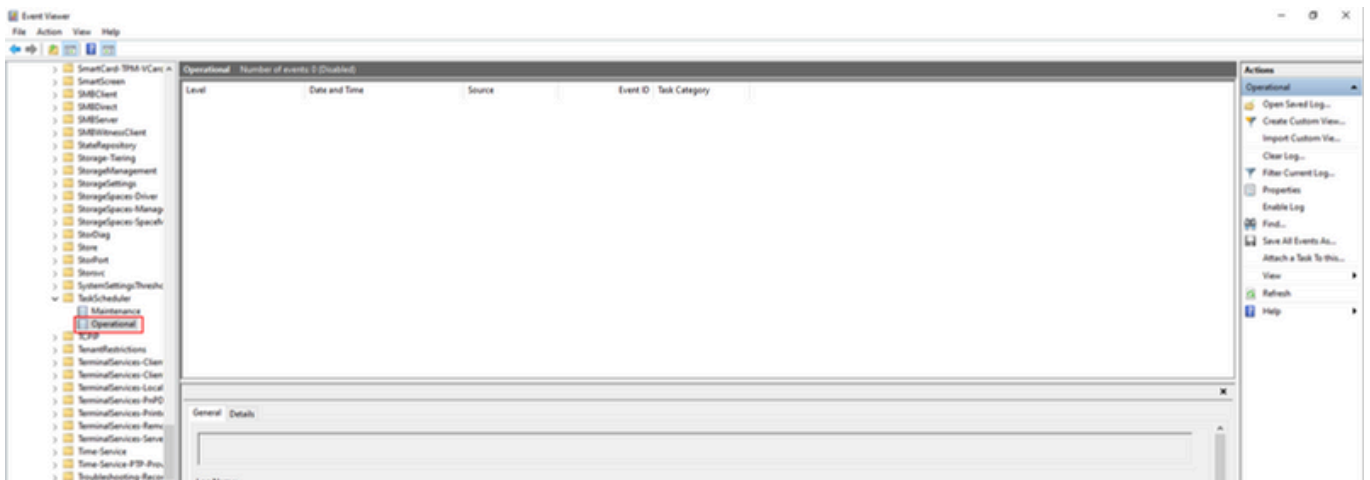
Ingesting Exported Windows Event Log Files

This guide shows how to collect Windows Task Scheduler logs and send them to Splunk using the Universal Forwarder.

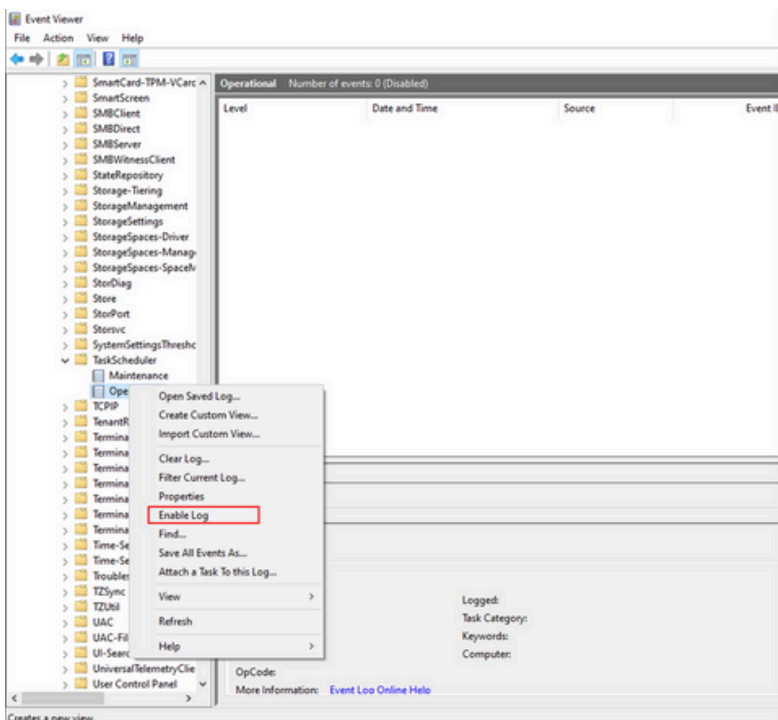
1. Enable Task Scheduler Logs

To collect detailed logs from Task Scheduler:

- Open Event Viewer (search for it in Windows).
- Go to: Applications and Services Logs > Microsoft > Windows > TaskScheduler



- Right-click **Operational** and choose **Enable Log**.

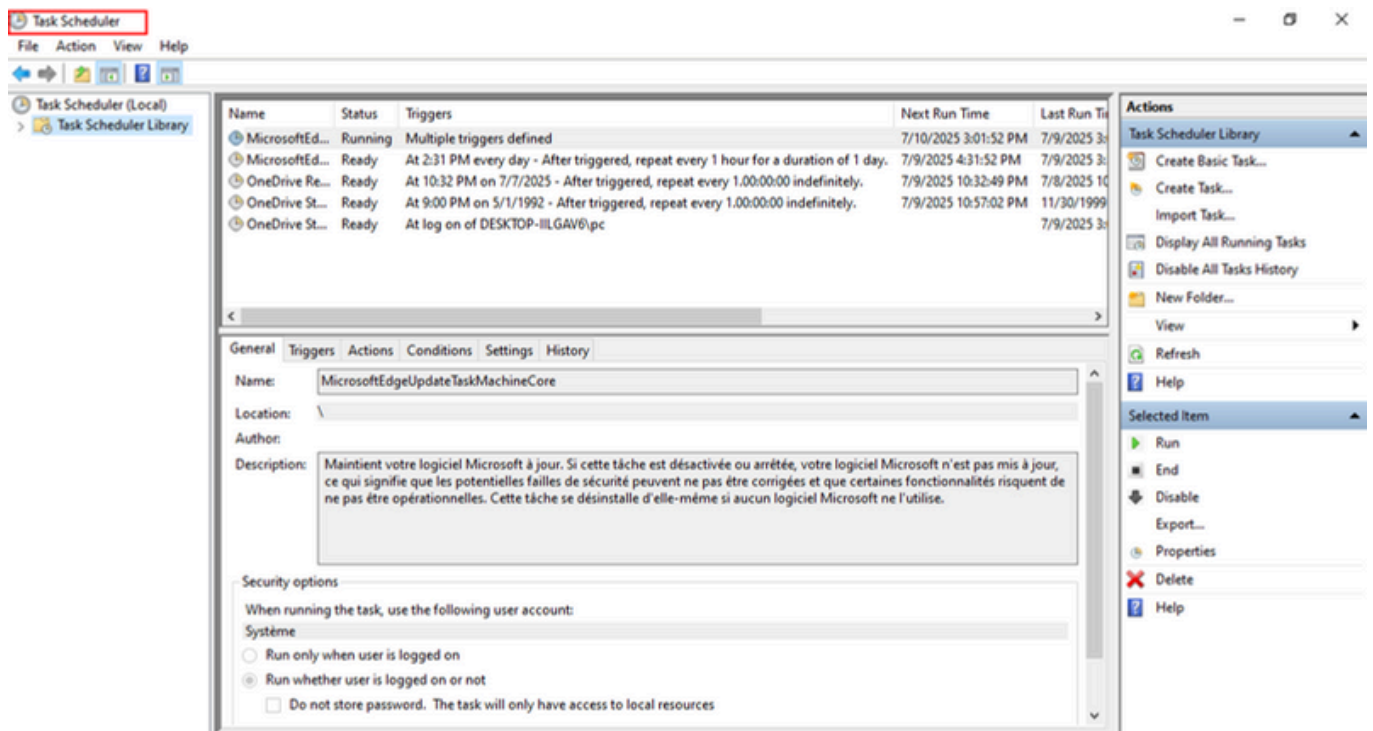


2. Simulate Persistence with a Scheduled Task

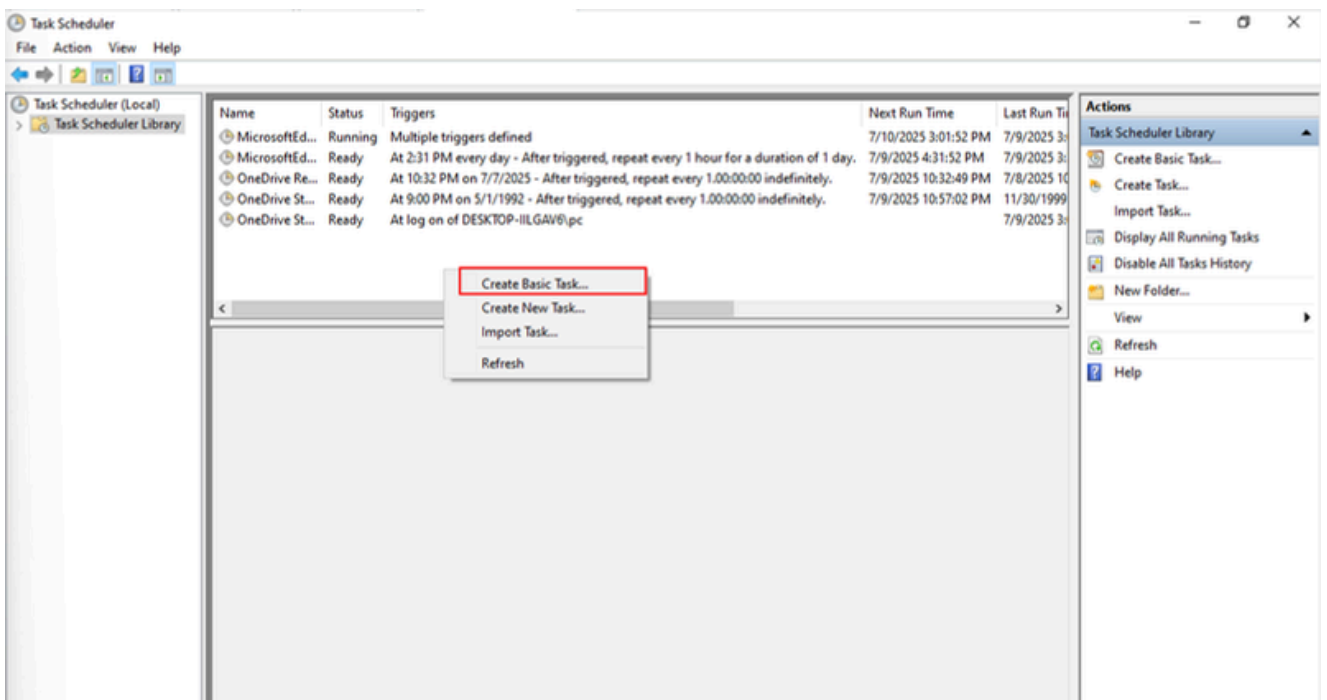
This step creates a fake scheduled task to simulate persistence (like malware staying active).

Steps:

1. Open Task Scheduler.

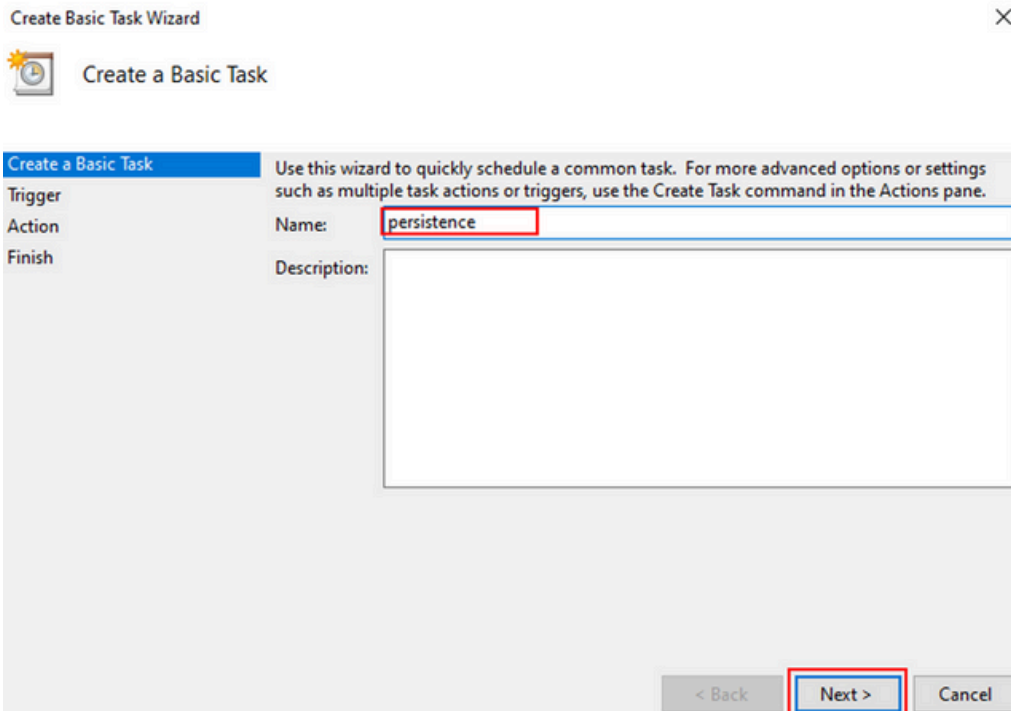


2. Click **Create Basic Task**.



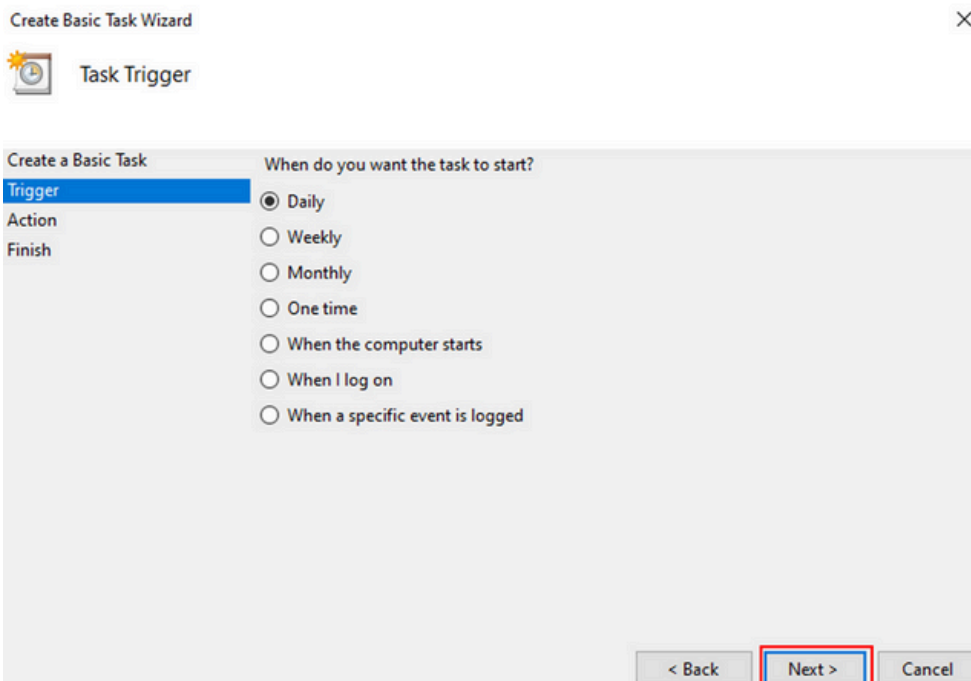
3. In the wizard:

- **Name:** persistence



The screenshot shows the 'Create Basic Task Wizard' dialog box. The title bar says 'Create Basic Task Wizard' with a close button. Below the title bar is a tab labeled 'Create a Basic Task'. The main area has a left sidebar with 'Trigger', 'Action', and 'Finish'. The 'Trigger' tab is selected. The main content area says: 'Use this wizard to quickly schedule a common task. For more advanced options or settings such as multiple task actions or triggers, use the Create Task command in the Actions pane.' Below this, there is a 'Name:' label followed by a text box containing 'persistence'. Below that is a 'Description:' label followed by a large empty text box. At the bottom right, there are three buttons: '< Back', 'Next >', and 'Cancel'. The 'Next >' button is highlighted with a red box.

- **Trigger:** Daily



The screenshot shows the 'Create Basic Task Wizard' dialog box, Step 2: Task Trigger. The title bar says 'Create Basic Task Wizard' with a close button. Below the title bar is a tab labeled 'Task Trigger'. The main area has a left sidebar with 'Trigger', 'Action', and 'Finish'. The 'Trigger' tab is selected. The main content area says: 'When do you want the task to start?'. Below this, there are seven radio button options: 'Daily' (selected), 'Weekly', 'Monthly', 'One time', 'When the computer starts', 'When I log on', and 'When a specific event is logged'. At the bottom right, there are three buttons: '< Back', 'Next >', and 'Cancel'. The 'Next >' button is highlighted with a red box.

- **Start time:** Choose any time

Create Basic Task Wizard ×

 Daily

Create a Basic Task

Trigger

Start: 7/ 9/2025 4:27:15 PM ☐ Synchronize across time zones

Recur every: 1 days

Daily

Action

Finish

< Back Next > Cancel

- **Action:** Start a program

Create Basic Task Wizard ×

 Action

Create a Basic Task

Trigger

Daily

Action

Finish

What action do you want the task to perform?

☒ Start a program

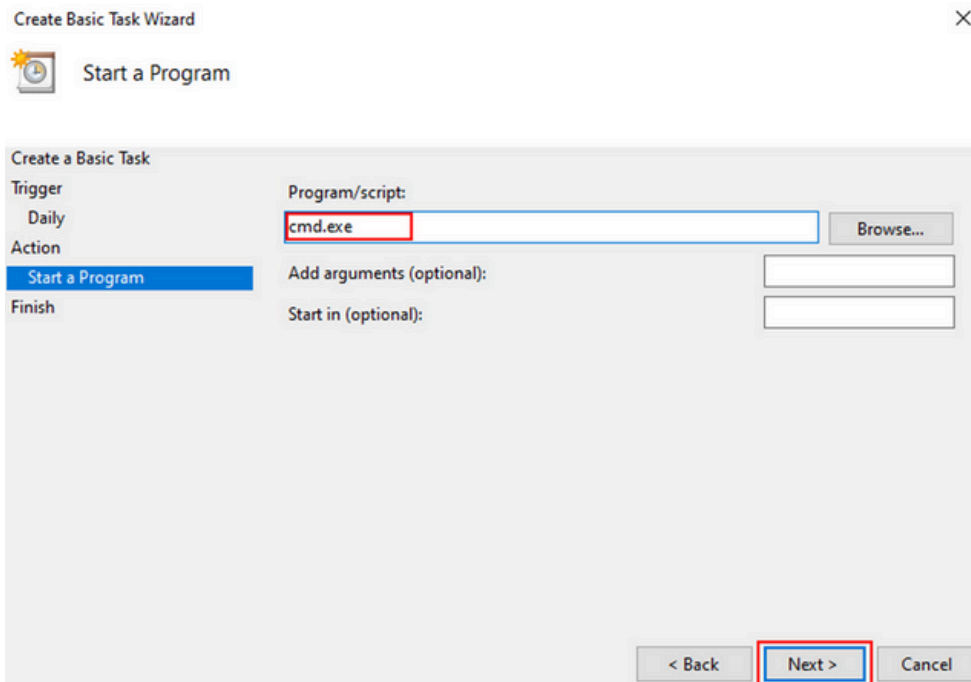
☐ Send an e-mail (deprecated)

☐ Display a message (deprecated)

< Back Next > Cancel

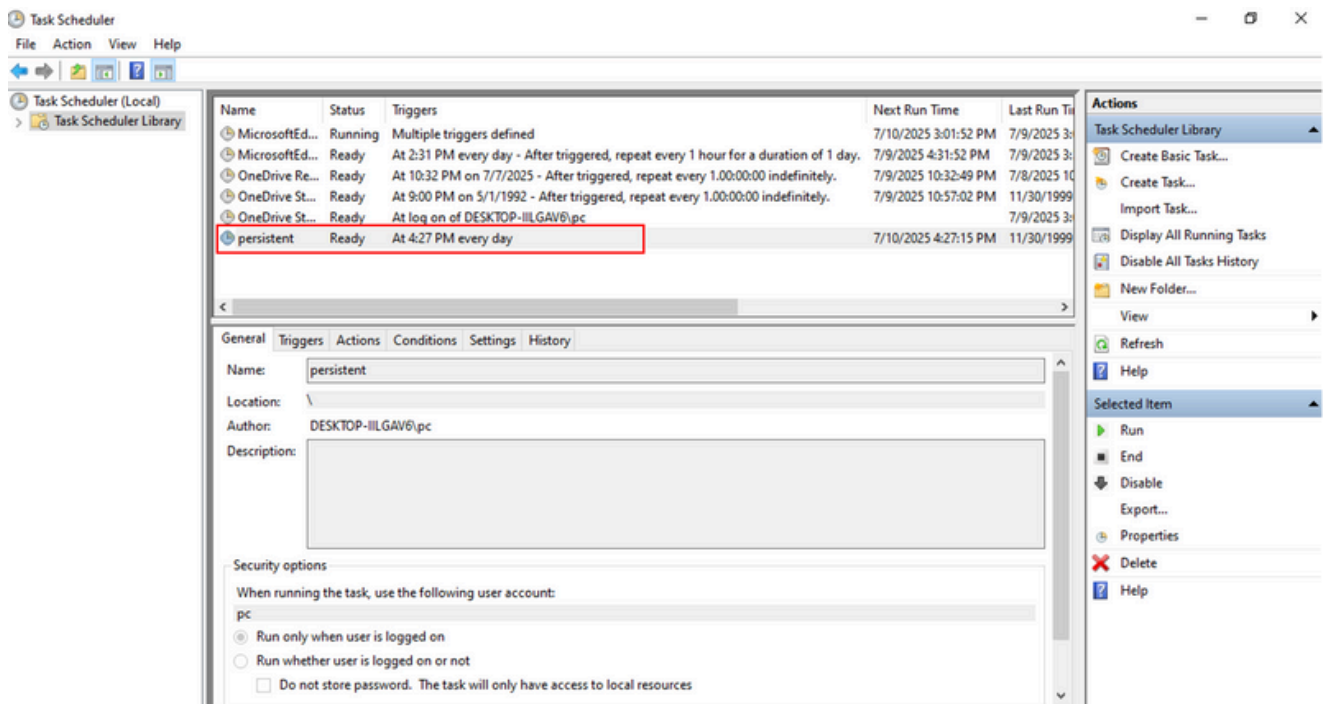
- **Program/script:** Type cmd.exe

Getting External Data into Splunk



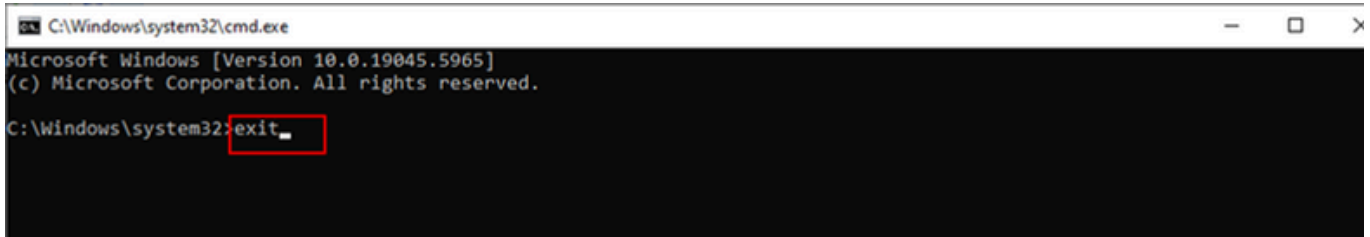
4. Click **Finish**.

5. You'll see the task listed as **Ready**.

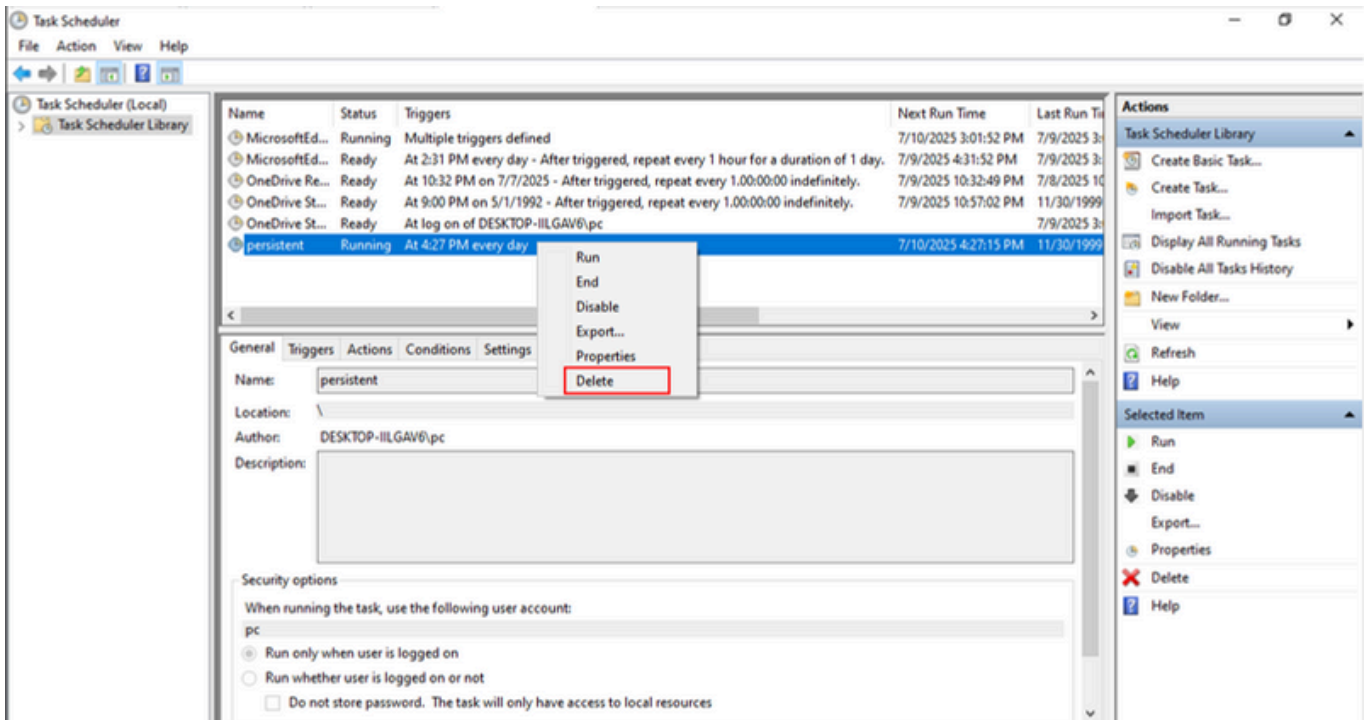


6. To test it, right-click the task and choose **Run**, a command prompt should open.

Getting External Data into Splunk



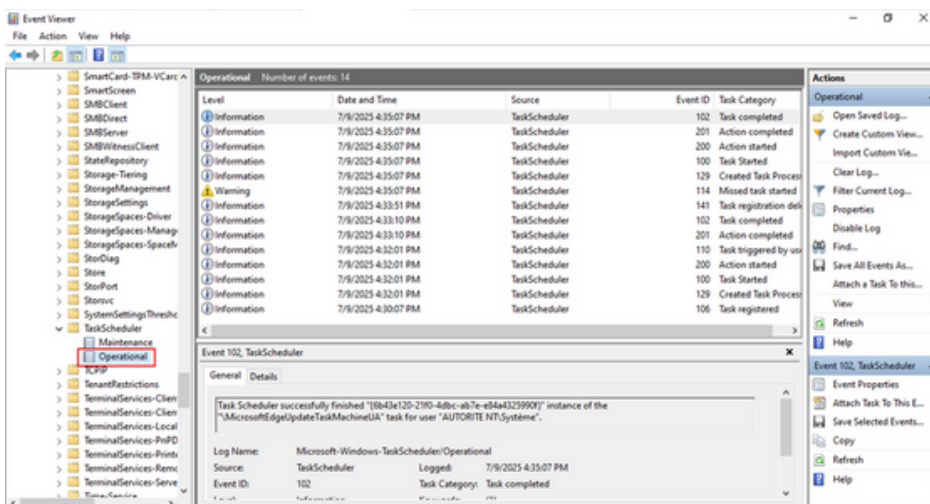
7. After testing, you can delete the task.



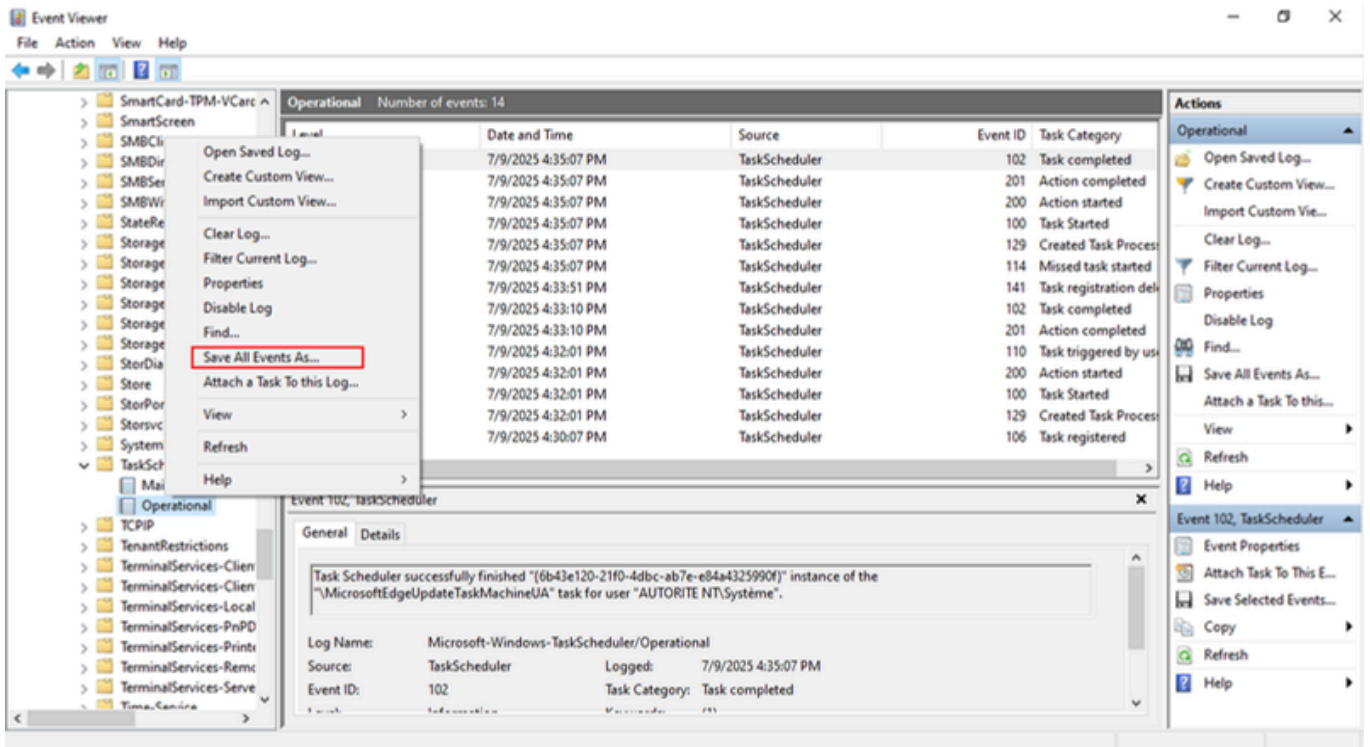
3. Save Task Scheduler Logs

1. Go back to Event Viewer →

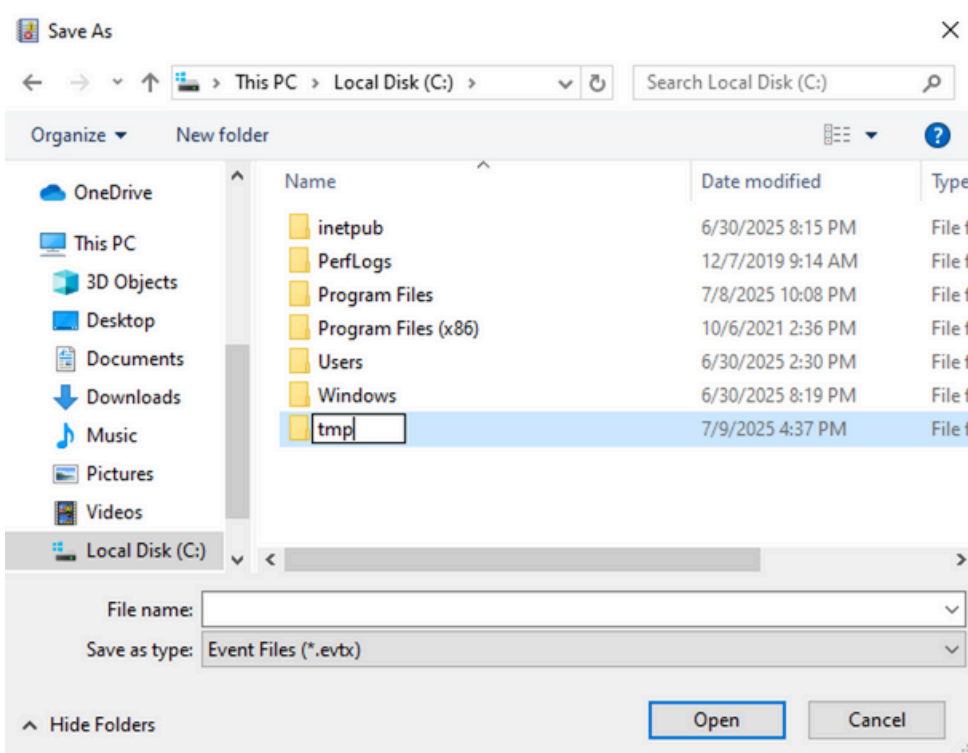
- Applications and Services Logs > Microsoft > Windows > TaskScheduler > Operational



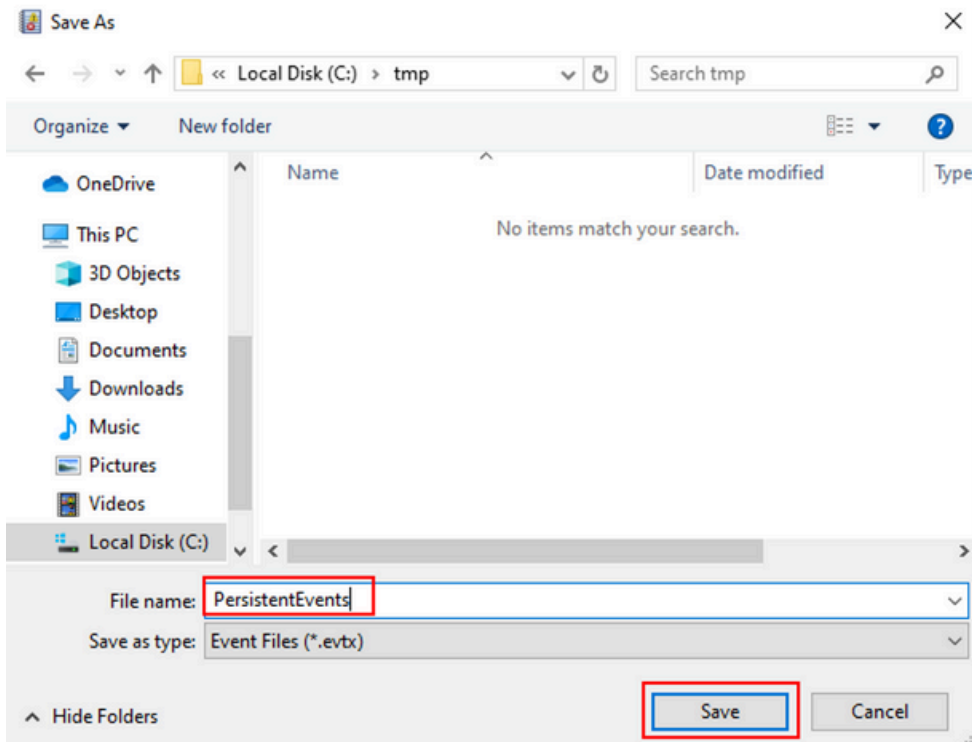
2. Right-click **Operational** → **Save All Events As...**



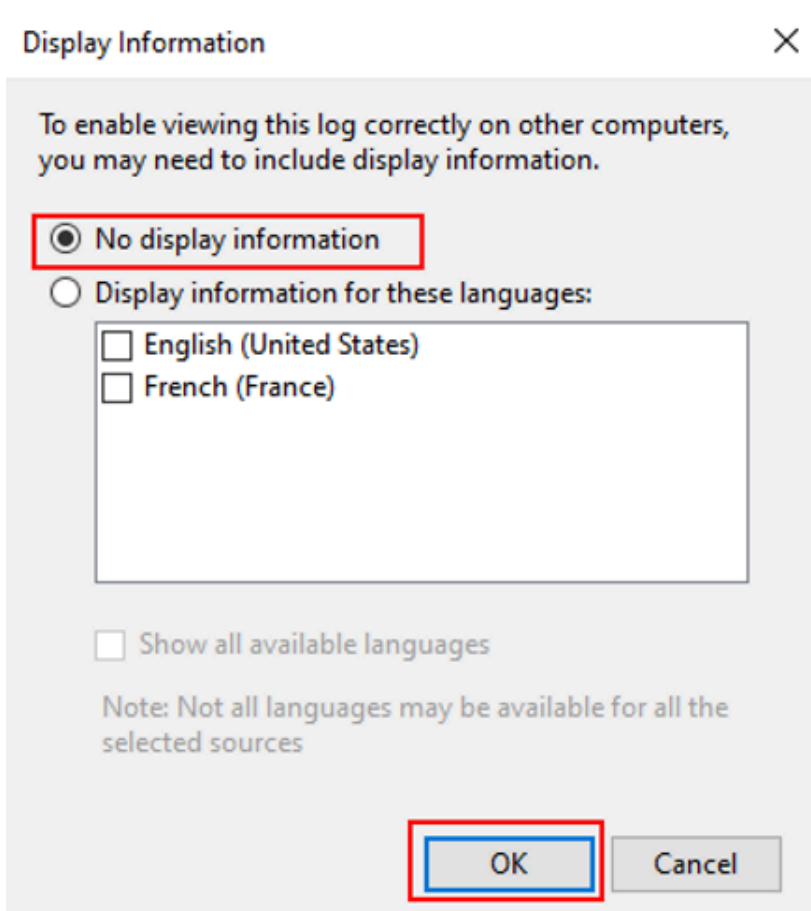
3. Choose a folder like C:\tmp



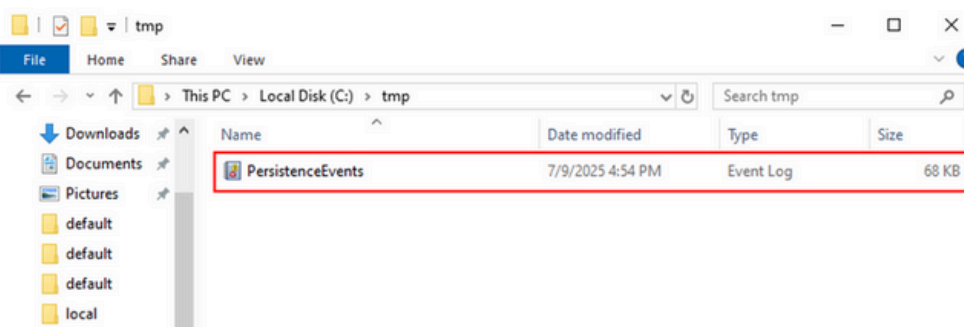
- Name the file: PersistentEvents.evtx
- Choose **Event Files (*.evtx)** as the file type
- Click **Save**



6. When asked about display info, choose **No display information**



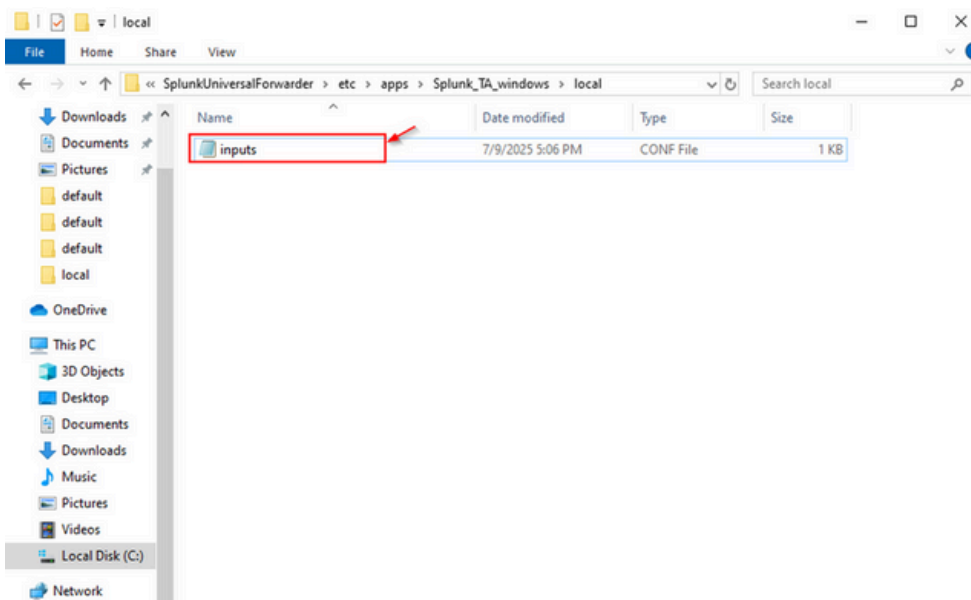
7. Confirm the file is saved in C:\tmp



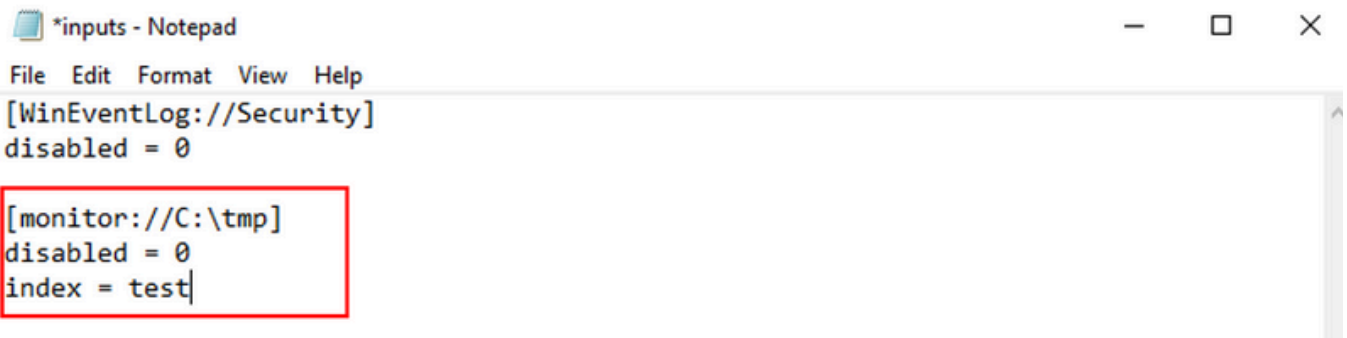
4. Configure Splunk Universal Forwarder to Monitor the Folder

1. Open the file:

- C:\Program Files\SplunkUniversalForwarder\etc\apps\Splunk_TA_windows\local\inputs.conf



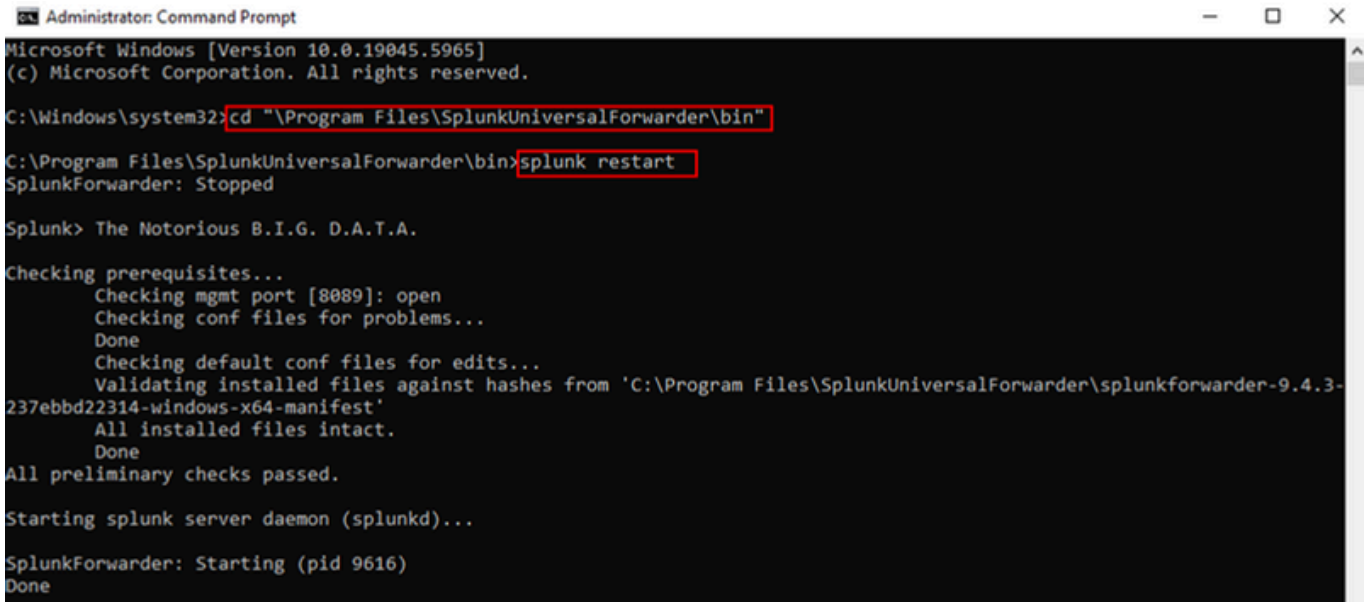
2. Add this configuration:



- disabled = 0 means monitoring is active
- index = test sends the data to the index named test

3. Restart the Universal Forwarder:

- Open **Admin Command Prompt**
- Go to: C:\Program Files\SplunkUniversalForwarder\bin
- Run: splunk restart



```
Administrator: Command Prompt
Microsoft Windows [Version 10.0.19045.5965]
(c) Microsoft Corporation. All rights reserved.

C:\Windows\system32>cd "\Program Files\SplunkUniversalForwarder\bin"

C:\Program Files\SplunkUniversalForwarder\bin>splunk restart
SplunkForwarder: Stopped

Splunk> The Notorious B.I.G. D.A.T.A.

Checking prerequisites...
  Checking mgmt port [8089]: open
  Checking conf files for problems...
  Done
  Checking default conf files for edits...
  Validating installed files against hashes from 'C:\Program Files\SplunkUniversalForwarder\splunkforwarder-9.4.3-237ebbd22314-windows-x64-manifest'
  All installed files intact.
  Done
All preliminary checks passed.

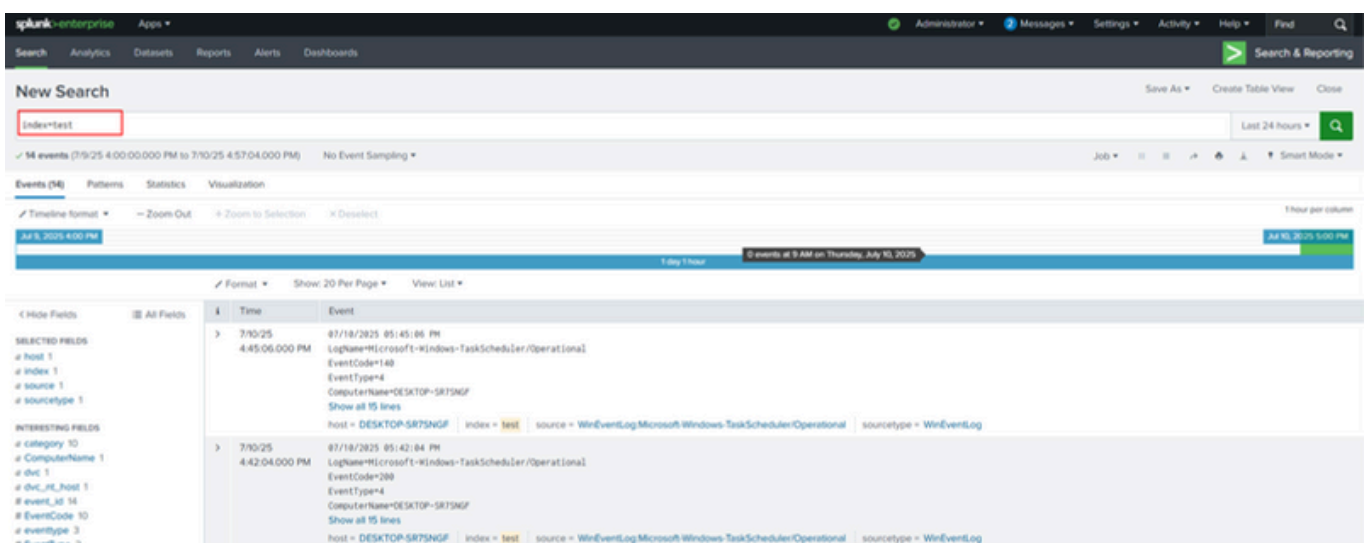
Starting splunk server daemon (splunkd)...

SplunkForwarder: Starting (pid 9616)
Done
```

5. Verify Logs in Splunk Web

To check if logs are coming in:

1. Open Splunk Web in your browser.
2. In the search bar, type: index=test
3. You should see events from the folder C:\tmp, like: WinEventLog:Microsoft-Windows-TaskScheduler/Operational



Onboarding Windows Registry Data

Monitoring the Windows Registry provides valuable insights for numerous use cases, aiding in security investigations and system health monitoring.

Why Monitor the Windows Registry?

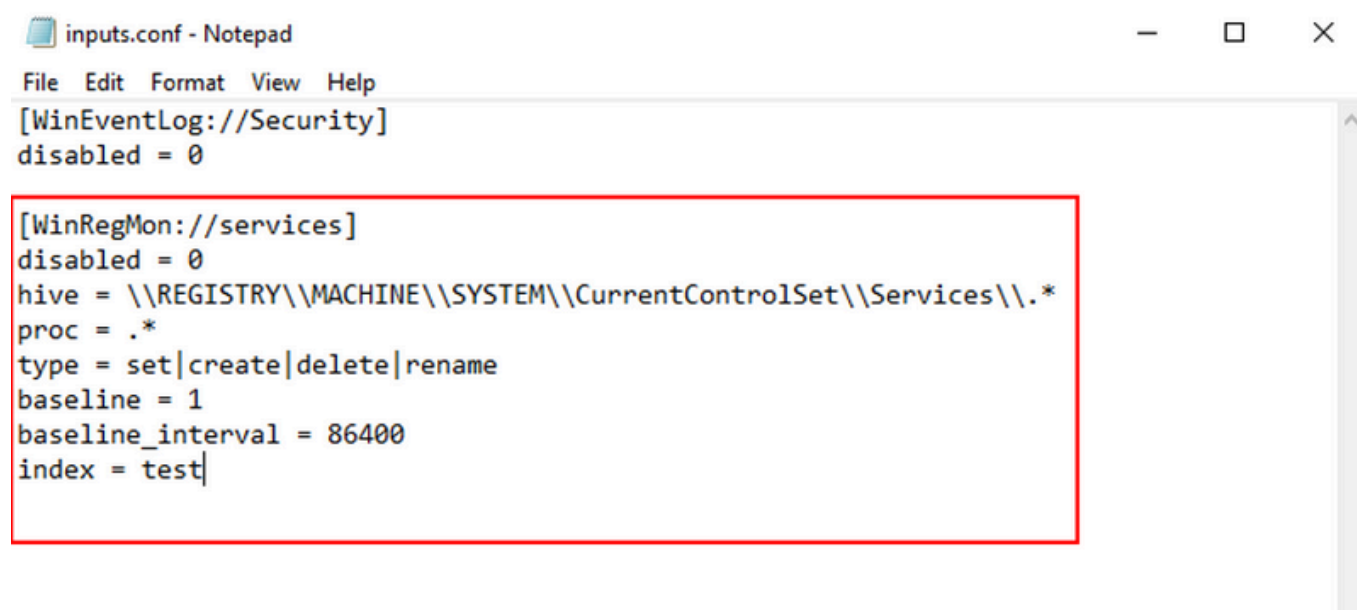
The Windows Registry is like a big settings database for Windows and its apps. Monitoring it helps you:

- **Detect new services:** Can show if malware or unknown software was installed.
- **Track autoruns:** Malware often sets itself to start with Windows.
- **Spot disabled security features:** Like WDigest being turned off — a sign of attack.
- **Help investigations:** Shows info about installed software, COM objects, etc.
- **Find weak settings:** Bad registry configs can be risky.
- **Detect unknown devices:** Registry logs can show if someone plugged in unauthorized hardware.

Configure Splunk to Monitor Registry

To collect registry data, you need to edit the inputs.conf file on the Windows machine where Splunk Universal Forwarder is installed.

- **Configuration**



The screenshot shows a Notepad window titled "inputs.conf - Notepad". The menu bar includes "File", "Edit", "Format", "View", and "Help". The text content of the file is as follows:

```
[WinEventLog://Security]
disabled = 0

[WinRegMon://services]
disabled = 0
hive = \\REGISTRY\\MACHINE\\SYSTEM\\CurrentControlSet\\Services\\.*
proc = .*
type = set|create|delete|rename
baseline = 1
baseline_interval = 86400
index = test|
```

The configuration for the [WinRegMon://services] section is highlighted with a red rectangular box.

What Each Line Means:

- WinRegMon://services: Just a name for this config.
- hive: Which part of the registry to watch, here it's services.
- proc: Watch all processes making changes.
- type: What kind of changes to track:
 - set: Change a value
 - create: Make a new key/value
 - delete: Remove a key/value
 - rename: Rename a key
- baseline = 1: Helps detect changes from a clean state.
- baseline_interval = 86400: Re-check every 24 hours.
- index = test: Send the data to the test index in Splunk.

Where to Put This Config

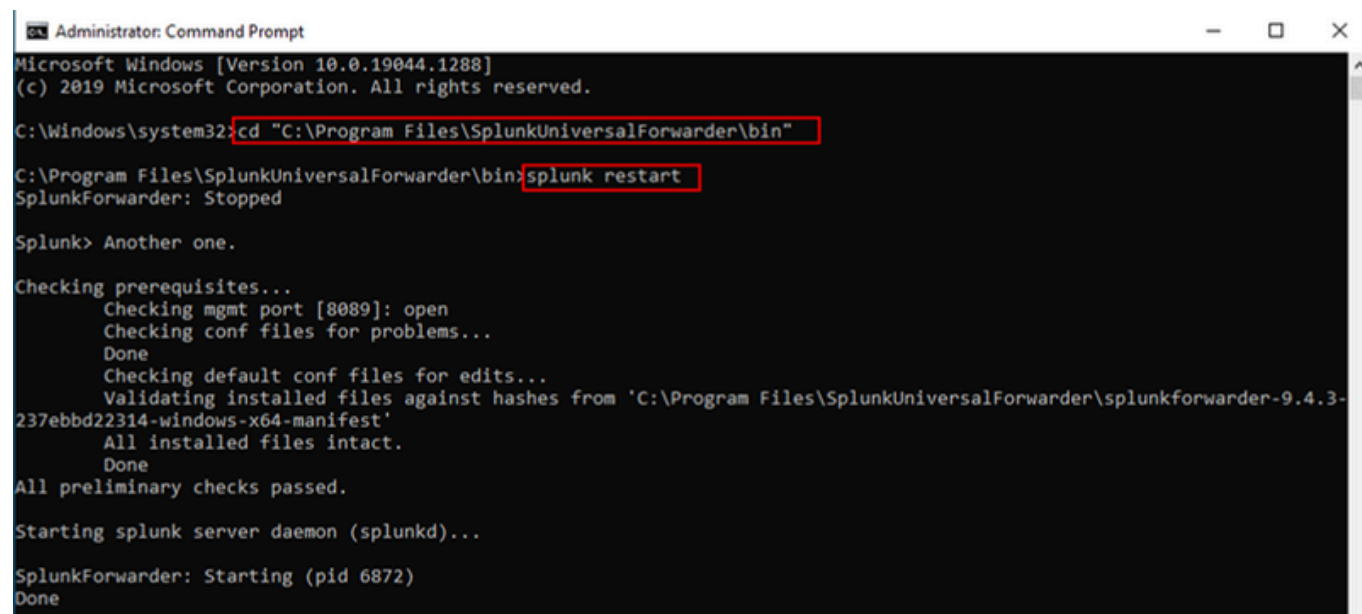
- File path:

C:\Program

Files\SplunkUniversalForwarder\etc\apps\Splunk_TA_windows\local\inputs.conf

Restart

After editing, restart Splunk Universal Forwarder:



```
Administrator: Command Prompt
Microsoft Windows [Version 10.0.19044.1288]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Windows\system32>cd "C:\Program Files\SplunkUniversalForwarder\bin"

C:\Program Files\SplunkUniversalForwarder\bin>splunk restart
SplunkForwarder: Stopped

Splunk> Another one.

Checking prerequisites...
  Checking mgmt port [8089]: open
  Checking conf files for problems...
  Done
  Checking default conf files for edits...
  Validating installed files against hashes from 'C:\Program Files\SplunkUniversalForwarder\splunkforwarder-9.4.3-237ebbd22314-windows-x64-manifest'
  All installed files intact.
  Done
All preliminary checks passed.

Starting splunk server daemon (splunkd)...

SplunkForwarder: Starting (pid 6872)
Done
```

Search and Analyze Registry Data in Splunk

Once data is coming in, you can search it in Splunk Web.

Basic Search:

The screenshot shows the Splunk Web interface with a search bar containing 'index=test'. The search results show 1,896 events. The interface includes a sidebar with 'SELECTED FIELDS' and 'INTERESTING FIELDS', a main search bar, and a results table. The results table shows two events with details like 'Time', 'Event', 'host', 'index', 'source', and 'sourcetype'.

Time	Event
7/11/2025 3:02:39.000 PM	event_status="(0)l'opération a réussi." pid=2316 process_image="vntoolsd.exe" registry_type="CreateKey" Show all 8 lines
7/11/2025 3:02:38.000 PM	event_status="(0)l'opération a réussi." pid=2316 process_image="vntoolsd.exe" registry_type="CreateKey" Show all 8 lines

Filter for Registry Events:

The screenshot shows the Splunk Web interface with a search bar containing 'index=test sourcetype=WinRegistry'. The search results show 1,940 events. The interface includes a sidebar with 'SELECTED FIELDS' and 'INTERESTING FIELDS', a main search bar, and a results table. The results table shows two events with details like 'Time', 'Event', 'host', 'index', 'source', and 'sourcetype'.

Time	Event
7/11/2025 3:11:09.000 PM	event_status="(0)l'opération a réussi." pid=2316 process_image="vntoolsd.exe" registry_type="CreateKey" Show all 8 lines
7/11/2025 3:10:54.000 PM	event_status="(0)l'opération a réussi." pid=3660 process_image="c:\Program Files (x86)\Microsoft\Edge\Application\msedge.exe" registry_type="SetValue" Show all 8 lines

If you don't see results, try changing the time range or check if sampling is enabled.

Time	Event																																																																																																																										
	<table><tr><th>Type</th><th>Field</th><th>Value</th><th>Actions</th></tr><tr><td rowspan="4">Selected</td><td>☒ host ▾</td><td>DESKTOP-SR7SNGF</td><td>▾</td></tr><tr><td>☒ index ▾</td><td>test</td><td>▾</td></tr><tr><td>☒ source ▾</td><td>WinRegistry</td><td>▾</td></tr><tr><td>☒ sourcetype ▾</td><td>WinRegistry</td><td>▾</td></tr><tr><td rowspan="36">Event</td><td>☐ action ▾</td><td>created</td><td>▾</td></tr><tr><td>☐ data ▾</td><td></td><td>▾</td></tr><tr><td>☐ data_type ▾</td><td>REG_NONE</td><td>▾</td></tr><tr><td>☐ dest ▾</td><td>DESKTOP-SR7SNGF</td><td>▾</td></tr><tr><td>☐ event_status ▾</td><td>{0}L'opération a réussi.</td><td>▾</td></tr><tr><td>☐ eventtype ▾</td><td>windows_ta_data</td><td>▾</td></tr><tr><td></td><td>winregistry_windows (change endpoint os registry windows)</td><td>▾</td></tr><tr><td>☐ key_path ▾</td><td>HKLM\system\currentcontrolset\services\tcpip\parameters</td><td>▾</td></tr><tr><td>☐ msg ▾</td><td>L'opération a réussi.</td><td>▾</td></tr><tr><td>☐ object ▾</td><td>parameters</td><td>▾</td></tr><tr><td>☐ object_category ▾</td><td>registry</td><td>▾</td></tr><tr><td>☐ object_path ▾</td><td>HKLM\system\currentcontrolset\services\tcpip\parameters</td><td>▾</td></tr><tr><td>☐ pid ▾</td><td>2316</td><td>▾</td></tr><tr><td>☐ process_id ▾</td><td>2316</td><td>▾</td></tr><tr><td>☐ process_image ▾</td><td>vmtoolsd.exe</td><td>▾</td></tr><tr><td>☐ registry_key_name ▾</td><td>parameters</td><td>▾</td></tr><tr><td>☐ registry_path ▾</td><td>HKLM\system\currentcontrolset\services\tcpip\parameters</td><td>▾</td></tr><tr><td>☐ registry_type ▾</td><td>CreateKey</td><td>▾</td></tr><tr><td>☐ registry_value_type ▾</td><td>REG_NONE</td><td>▾</td></tr><tr><td>☐ src ▾</td><td>DESKTOP-SR7SNGF</td><td>▾</td></tr><tr><td>☐ status ▾</td><td>success</td><td>▾</td></tr><tr><td>☐ tag ▾</td><td>change</td><td>▾</td></tr><tr><td></td><td>endpoint</td><td>▾</td></tr><tr><td></td><td>os</td><td>▾</td></tr><tr><td></td><td>registry</td><td>▾</td></tr><tr><td></td><td>windows</td><td>▾</td></tr><tr><td>☐ user_type ▾</td><td>process</td><td>▾</td></tr><tr><td>☐ vendor_action ▾</td><td>CreateKey</td><td>▾</td></tr><tr><td>☐ vendor_product ▾</td><td>Microsoft Windows</td><td>▾</td></tr><tr><td>☐ vendor_status ▾</td><td>0</td><td>▾</td></tr><tr><td>Time</td><td>_time ▾</td><td>2025-07-11T15:11:09.000+00:00</td><td></td></tr><tr><td rowspan="3">Default</td><td>☐ linecount ▾</td><td>8</td><td>▾</td></tr><tr><td>☐ punct ▾</td><td>//_="{} _""=*"~%~\ / ~""=*</td><td>▾</td></tr><tr><td>☐ splunk_server ▾</td><td>splunk</td><td>▾</td></tr></table>	Type	Field	Value	Actions	Selected	☒ host ▾	DESKTOP-SR7SNGF	▾	☒ index ▾	test	▾	☒ source ▾	WinRegistry	▾	☒ sourcetype ▾	WinRegistry	▾	Event	☐ action ▾	created	▾	☐ data ▾		▾	☐ data_type ▾	REG_NONE	▾	☐ dest ▾	DESKTOP-SR7SNGF	▾	☐ event_status ▾	{0}L'opération a réussi.	▾	☐ eventtype ▾	windows_ta_data	▾		winregistry_windows (change endpoint os registry windows)	▾	☐ key_path ▾	HKLM\system\currentcontrolset\services\tcpip\parameters	▾	☐ msg ▾	L'opération a réussi.	▾	☐ object ▾	parameters	▾	☐ object_category ▾	registry	▾	☐ object_path ▾	HKLM\system\currentcontrolset\services\tcpip\parameters	▾	☐ pid ▾	2316	▾	☐ process_id ▾	2316	▾	☐ process_image ▾	vmtoolsd.exe	▾	☐ registry_key_name ▾	parameters	▾	☐ registry_path ▾	HKLM\system\currentcontrolset\services\tcpip\parameters	▾	☐ registry_type ▾	CreateKey	▾	☐ registry_value_type ▾	REG_NONE	▾	☐ src ▾	DESKTOP-SR7SNGF	▾	☐ status ▾	success	▾	☐ tag ▾	change	▾		endpoint	▾		os	▾		registry	▾		windows	▾	☐ user_type ▾	process	▾	☐ vendor_action ▾	CreateKey	▾	☐ vendor_product ▾	Microsoft Windows	▾	☐ vendor_status ▾	0	▾	Time	_time ▾	2025-07-11T15:11:09.000+00:00		Default	☐ linecount ▾	8	▾	☐ punct ▾	//_="{} _""=*"~%~\ / ~""=*	▾	☐ splunk_server ▾	splunk	▾
Type	Field	Value	Actions																																																																																																																								
Selected	☒ host ▾	DESKTOP-SR7SNGF	▾																																																																																																																								
	☒ index ▾	test	▾																																																																																																																								
	☒ source ▾	WinRegistry	▾																																																																																																																								
	☒ sourcetype ▾	WinRegistry	▾																																																																																																																								
Event	☐ action ▾	created	▾																																																																																																																								
	☐ data ▾		▾																																																																																																																								
	☐ data_type ▾	REG_NONE	▾																																																																																																																								
	☐ dest ▾	DESKTOP-SR7SNGF	▾																																																																																																																								
	☐ event_status ▾	{0}L'opération a réussi.	▾																																																																																																																								
	☐ eventtype ▾	windows_ta_data	▾																																																																																																																								
		winregistry_windows (change endpoint os registry windows)	▾																																																																																																																								
	☐ key_path ▾	HKLM\system\currentcontrolset\services\tcpip\parameters	▾																																																																																																																								
	☐ msg ▾	L'opération a réussi.	▾																																																																																																																								
	☐ object ▾	parameters	▾																																																																																																																								
	☐ object_category ▾	registry	▾																																																																																																																								
	☐ object_path ▾	HKLM\system\currentcontrolset\services\tcpip\parameters	▾																																																																																																																								
	☐ pid ▾	2316	▾																																																																																																																								
	☐ process_id ▾	2316	▾																																																																																																																								
	☐ process_image ▾	vmtoolsd.exe	▾																																																																																																																								
	☐ registry_key_name ▾	parameters	▾																																																																																																																								
	☐ registry_path ▾	HKLM\system\currentcontrolset\services\tcpip\parameters	▾																																																																																																																								
	☐ registry_type ▾	CreateKey	▾																																																																																																																								
	☐ registry_value_type ▾	REG_NONE	▾																																																																																																																								
	☐ src ▾	DESKTOP-SR7SNGF	▾																																																																																																																								
	☐ status ▾	success	▾																																																																																																																								
	☐ tag ▾	change	▾																																																																																																																								
		endpoint	▾																																																																																																																								
		os	▾																																																																																																																								
		registry	▾																																																																																																																								
		windows	▾																																																																																																																								
	☐ user_type ▾	process	▾																																																																																																																								
	☐ vendor_action ▾	CreateKey	▾																																																																																																																								
	☐ vendor_product ▾	Microsoft Windows	▾																																																																																																																								
	☐ vendor_status ▾	0	▾																																																																																																																								
	Time	_time ▾	2025-07-11T15:11:09.000+00:00																																																																																																																								
	Default	☐ linecount ▾	8	▾																																																																																																																							
		☐ punct ▾	//_="{} _""=*"~%~\ / ~""=*	▾																																																																																																																							
		☐ splunk_server ▾	splunk	▾																																																																																																																							

What You'll See in Results

Each registry event will show:

- time: When it happened
- event_status: Was it successful?
- process_image: Which program made the change
- registry_type: Type of change (e.g., createKey, SetValue)
- host: Which machine
- source: Where the data came from
- sourcetype: Should be WinRegistry
- action: What happened (e.g., modified)
- key_path: Full path to the registry key
- registry_key_name: Name of the key
- registry_path: Path to the key
- registry_value_name: Name of the value
- registry_value_type: Type of data (e.g., REG_BINARY)

Ingesting Dynamic Windows Registry Data

Splunk's powerful data ingestion features allow organizations to collect and analyze machine data for security, operations, and performance. Going beyond basic log collection, a strategic approach focuses on targeted data to reduce noise and improve insights.

Strategic Data Collection Using `inputs.conf`

The `inputs.conf` file on Splunk Universal Forwarders is where you define what data to collect. It gives you fine control to:

- Filter specific events
- Reduce resource usage
- Focus on relevant security signals

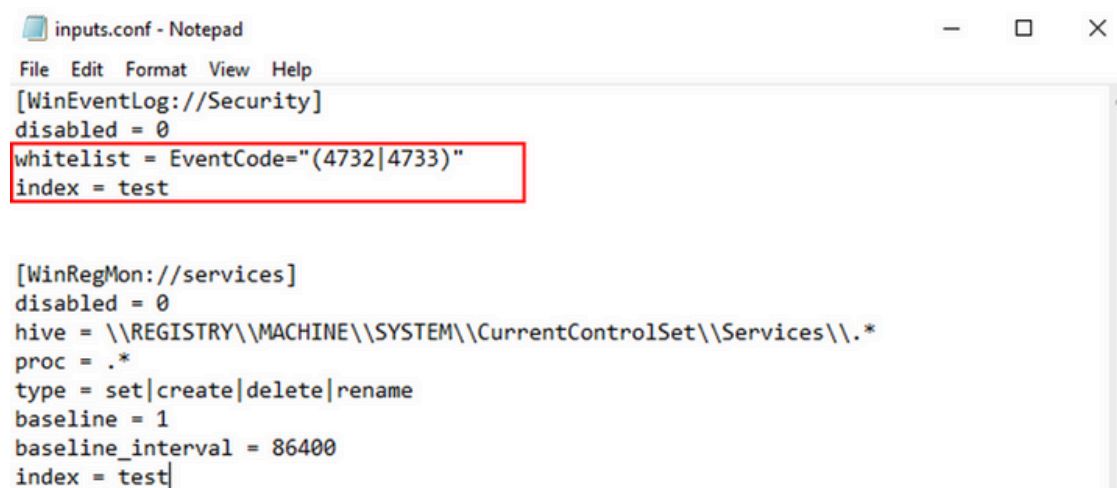
Example: Targeted Windows Security Event Log Ingestion

Instead of collecting all logs, you can focus on critical security events, like changes to security-enabled groups (e.g., local Administrators group).

Key Event IDs:

- 4732: A user was added to a security-enabled local group
- 4733: A user was removed from a security-enabled local group

Sample Configuration in `inputs.conf`:



```
inputs.conf - Notepad
File Edit Format View Help
[WinEventLog://Security]
disabled = 0
whitelist = EventCode="(4732|4733)"
index = test

[WinRegMon://services]
disabled = 0
hive = \\REGISTRY\\MACHINE\\SYSTEM\\CurrentControlSet\\Services\\.*
proc = .*
type = set|create|delete|rename
baseline = 1
baseline_interval = 86400
index = test|
```

- `whitelist`: Only collects events with ID 4732 or 4733
- `index`: Sends data to the `security_monitoring` index
- After modifying `inputs.conf`, a **restart** of the Splunk Universal Forwarder is mandatory for the changes to take effect:

```
C:\Program Files\SplunkUniversalForwarder\bin>splunk restart
SplunkForwarder: Stopped

Splunk> Another one.

Checking prerequisites...
  Checking mgmt port [8089]: open
  Checking conf files for problems...
  Done
  Checking default conf files for edits...
  Validating installed files against hashes from 'C:\Program Files\SplunkUniversalForwarder\splunkforwarder-9.4.3-237ebbd22314-windows-x64-manifest'
  All installed files intact.
  Done
All preliminary checks passed.

Starting splunk server daemon (splunkd)...
SplunkForwarder: Starting (pid 4812)
Done
```

This method is called source-side filtering, it reduces bandwidth and indexing costs by filtering logs before they reach Splunk.

Verifying Ingestion

To test if it works:

1. Add/remove a user from the local Administrators group
2. Search in Splunk:

The screenshot shows the Splunk Search interface. The search bar contains the query `index=test sourcetype=XmlWinEventLog EventCode=4732 OR EventCode=4733`. The search results show 4 events. The first event is expanded, showing the following XML data:

```
<Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event"><System><Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-a5ba-3e3b0328c38d}" /><EventID>4732</EventID><Version>0</Version><Level>0</Level><Task>13826</Task><Opcode>0</Opcode><Keywords>0x8020000000000000</Keywords><TimeCreated SystemTime="2025-07-11T16:40:07.0559279Z"/><EventRecordID>11277</EventRecordID><Correlation ActivityID="{c030a583-f281-0001-e8a5-30c081f2db01}" /><Execution ProcessID="656" ThreadID="7104" /><Channel>Security</Channel><Computer>DESKTOP-SR7SNGF</Computer><Security></System><EventData><Data Name="MemberName"></Data><Data Name="MemberSid">DESKTOP-SR7SNGF\hack</Data><Data Name="TargetUserName">Utilisateurs</Data><Data Name="TargetDomainName">Builtin</Data><Data Name="TargetSid">BUILTIN\Utilisateurs</Data><Data Name="SubjectUserSid">DESKTOP-SR7SNGF\pc</Data><Data Name="SubjectUserName">pc</Data><Data Name="SubjectDomainName">DESKTOP-SR7SNGF</Data><Data Name="SubjectLogonId">0x39a32</Data><Data Name="PrivilegeList"></Data></EventData></Event>
```

The event details at the bottom show: EventCode = 4732, host = DESKTOP-SR7SNGF, index = test, source = XmlWinEventLog:Security, sourcetype = XmlWinEventLog.

Beyond EventCode: Using whitelist Flexibly

The whitelist (and its opposite, blacklist or denylist) can filter any field Splunk parses, not just EventCode.

This means you can:

- Target specific usernames
- Filter by process name
- Focus on registry paths, etc.

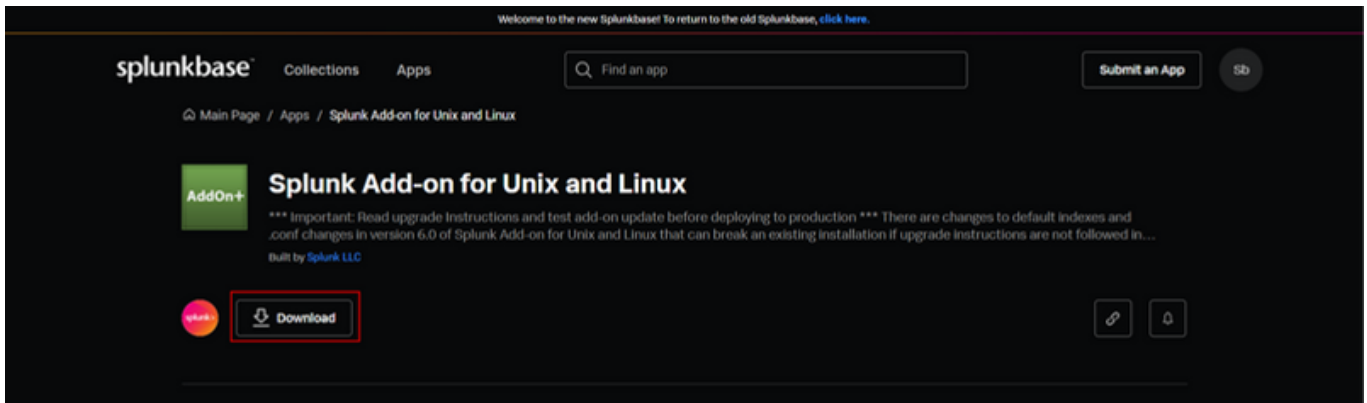
For advanced use, check Splunk's official documentation on allow/deny list formats.

Onboarding Generic Linux System Logs

To collect Linux logs (like /var/log/auth.log) and send them to Splunk, you need the Splunk Add-on for Unix and Linux installed on your Universal Forwarder (UF).

Step 1: Download the Add-on

- Go to Splunkbase and download **Splunk Add-on for Unix and Linux** (.tgz file).



Step 2: Transfer to the Universal Forwarder

- Use SCP to copy the .tgz file to your UF:

```
PS C:\WINDOWS\system32> scp C:\Users\pc\Downloads\splunk-add-on-for-unix-and-linux_1010.tgz ubuntu@192.168.1.15:/home/ubuntu/  
ubuntu@192.168.1.15's password:  
splunk-add-on-for-unix-and-linux_1010.tgz 100% 132KB 21.4MB/s 00:00
```

- Verify with:

```
ubuntu@ubuntu:~$ ls -l  
total 87100  
-rw-rw-r-- 1 ubuntu ubuntu 134865 Jul 12 16:26 splunk-add-on-for-unix-and-linux_1010.tgz  
drwxr-x--- 11 ubuntu ubuntu 4096 Jul 12 13:43 splunkforwarder  
-rw-rw-r-- 1 ubuntu ubuntu 89045711 Jun 4 18:47 splunkforwarder-9.4.3-237ebbd22314-linux-amd64.tgz
```

You should see the .tgz file.

Step 3: Install the Add-on

- Move the file to the apps directory:

```
ubuntu@ubuntu:~$ mv splunk-add-on-for-unix-and-linux_1010.tgz /home/ubuntu/splunkforwarder/etc/apps/
```

- Navigate to the directory and extract:

```
ubuntu@ubuntu:~$ cd ./splunkforwarder/etc/apps/  
ubuntu@ubuntu:~/splunkforwarder/etc/apps$ ls  
audit_trail learned splunk_httpinput  
introspection_generator_addon search splunk_internal_metrics  
journald_input splunk-add-on-for-unix-and-linux_1010.tgz SplunkUniversalForwarder
```

```
ubuntu@ubuntu:~/splunkforwarder/etc/apps$ tar -xf splunk-add-on-for-unix-and-linux_1010.tgz
ubuntu@ubuntu:~/splunkforwarder/etc/apps$ ls
audit_trail          search               Splunk_TA_nix
introspection_generator_addon  splunk-add-on-for-unix-and-linux_1010.tgz  SplunkUniversalForwarder
journald_input       splunk_httpinput
learned              splunk_internal_metrics
```

You should see Splunk_TA_nix.

- **Restart UF:**

```
ubuntu@ubuntu:~/splunkforwarder/bin$ sudo ./splunk restart
[sudo] password for ubuntu:
Warning: Attempting to revert the SPLUNK_HOME ownership
Warning: Executing "chown -R ubuntu:ubuntu /home/ubuntu/splunkforwarder"
splunkd is not running.
splunkd.pid doesn't exist...

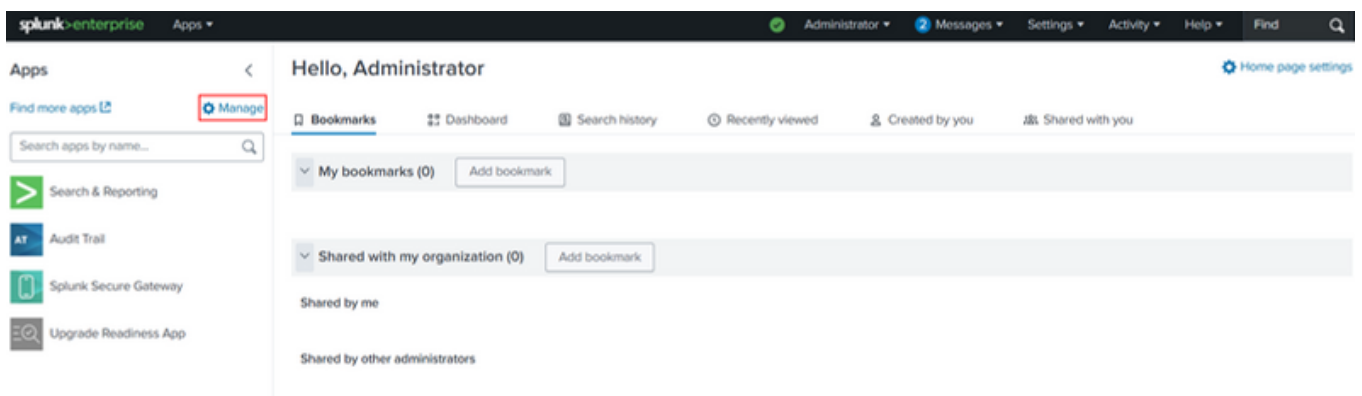
Splunk> The IT Search Engine.

Checking prerequisites...
  Checking mgmt port [8089]: open
  Checking conf files for problems...
  Done
  Checking default conf files for edits...
  Validating installed files against hashes from '/home/ubuntu/splunkforwarder/splunkforwarder
-9.4.3-237ebbd22314-linux-amd64-manifest'
  All installed files intact.
  Done
All preliminary checks passed.

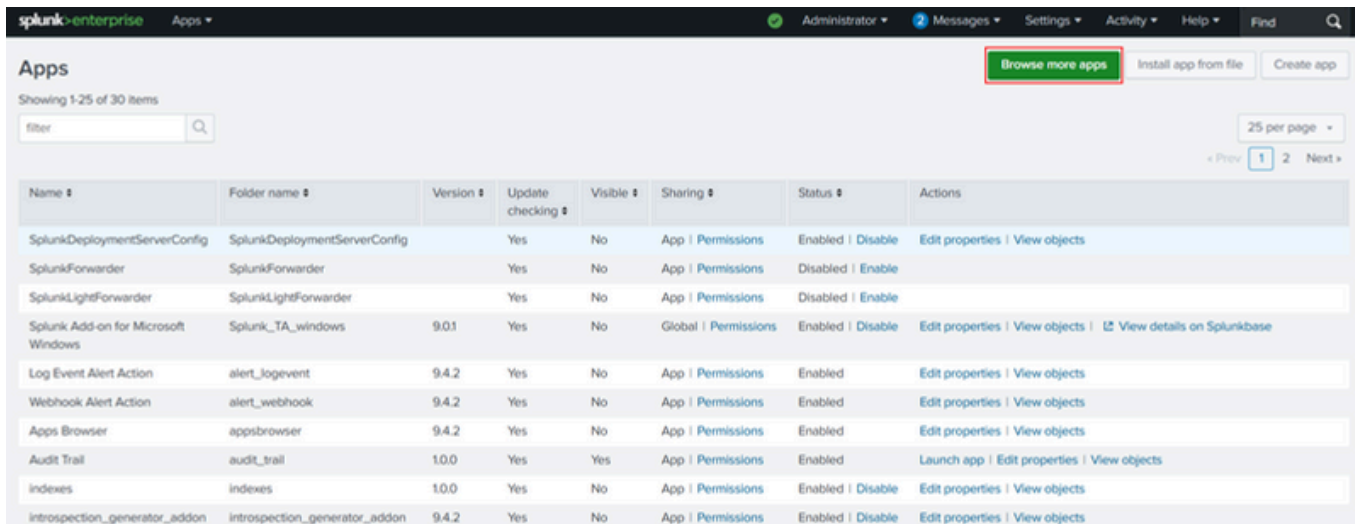
Starting splunk server daemon (splunkd)...
Done
```

4. Alternative Method (Splunk Web)

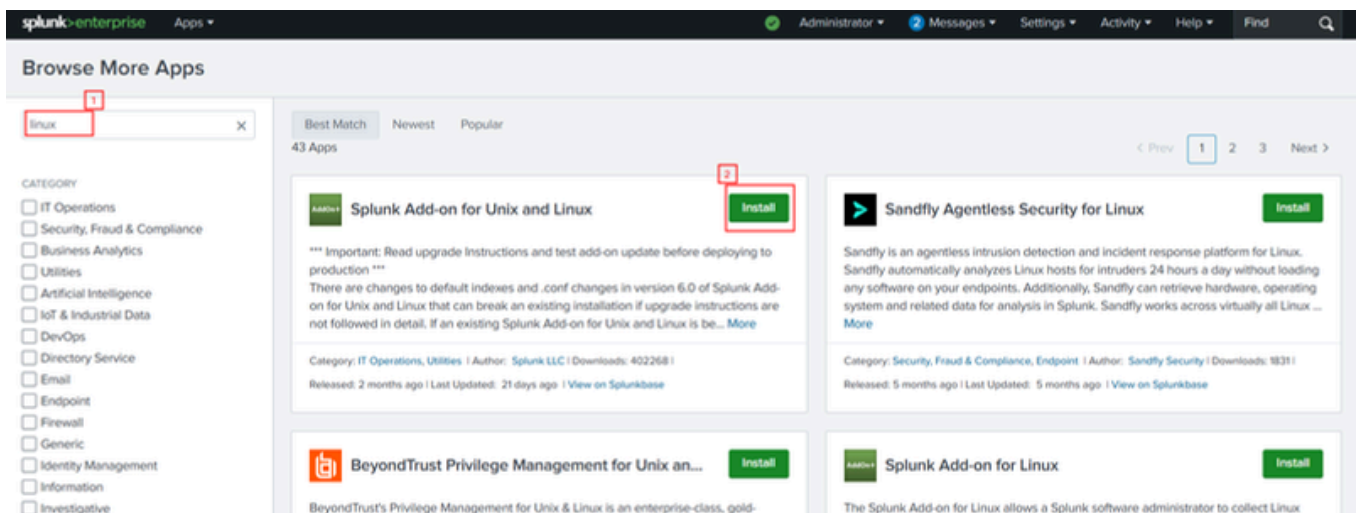
- Go to **Apps (Manage)** → **Browse More Apps** in Splunk Enterprise.



Getting External Data into Splunk



- Search for **Splunk Add-on for Unix and Linux** and click Install.



Step 4: Configure Data Input

- Create a local directory inside Splunk_TA_nix if it doesn't exist.

```
ubuntu@ubuntu:~/splunkforwarder$ cd ../etc/apps/
ubuntu@ubuntu:~/splunkforwarder/etc/apps$ ls
audit_trail          search               Splunk_TA_nix
introspection_generator_addon splunk-add-on-for-unix-and-linux_1010.tgz SplunkUniversalForwarder
journald_input       splunk_httpinput
learned              splunk_internal_metrics
ubuntu@ubuntu:~/splunkforwarder/etc/apps$ cd Splunk_TA_nix/
ubuntu@ubuntu:~/splunkforwarder/etc/apps/Splunk_TA_nix$ mkdir local`C`
```

- Create inputs.conf:

```
ubuntu@ubuntu:~/splunkforwarder/etc/apps/Splunk_TA_nix$ cd local/
ubuntu@ubuntu:~/splunkforwarder/etc/apps/Splunk_TA_nix/local$ nano inputs.conf
```

```
GNU nano 6.2                               inputs.conf *
[monitor:///var/log/auth.log]
sourcetype = linux_secure
index = test
```

- Save and exit.
- **Restart UF:**

```
ubuntu@ubuntu:~/splunkforwarder/bin$ sudo ./splunk restart
[sudo] password for ubuntu:
Warning: Attempting to revert the SPLUNK_HOME ownership
Warning: Executing "chown -R ubuntu:ubuntu /home/ubuntu/splunkforwarder"
Stopping splunkd...
Shutting down. Please wait, as this may take a few minutes.

Stopping splunk helpers...

Done.
splunkd.pid doesn't exist...

Splunk> The IT Search Engine.
```

6. Verifying Data Ingestion in Splunk Enterprise

After restarting the Splunk Universal Forwarder, you can confirm that authentication logs from your Linux system are being successfully sent to Splunk.

Step-by-Step Verification

1. Perform a Search

- Go to the Search & Reporting app
- In the search bar, enter:

The screenshot shows the Splunk Search & Reporting app interface. The search bar at the top contains the query `index=test`. Below the search bar, the results are displayed in a table view. The table has two columns: 'Time' and 'Event'. There are two events listed, both from 7/12/25 at 8:16:11.000 PM. The first event is a sudo session opening for user root. The second event is a splunk restart command. The search results are filtered by host=splunk, index=test, source=/var/log/auth.log, and sourcetype=linux_secure.

Time	Event
7/12/25 8:16:11.000 PM	Jul 12 20:16:11 splunk sudo: pam_unix(sudo:session): session opened for user root(uid=0) by splunk(uid=1000)
7/12/25 8:16:11.000 PM	Jul 12 20:16:11 splunk sudo: splunk : TTY=ttty1 ; PWD=/home/splunk/splunk/bin ; USER=root ; COMMAND=./splunk restart

2. Review Search Results

You should see events coming from `/var/log/auth.log`. These events confirm that your Universal Forwarder is working correctly.

Getting External Data into Splunk

Time

Event

7/12/25

8:16:11.000 PM

Jul 12 20:16:11 splunk sudo: pam_unix(sudo:session): session opened for user root(uid=0) by splunk(uid=1000)

Event Actions

Type	Field	Value	Actions
Selected	☒ host	splunk	
	☒ index	test	
	☒ source	/var/log/auth.log	
	☒ sourcetype	linux_secure	
Event	☐ action	success	
	☐ app	sudo (local privileged)	
	☐ authentication_service	pam_unix	
	☐ dest	splunk	
	☐ dvc	splunk	
	☐ eventtype	nix-all-logs	
		nix_security (os_unix)	
		nix_ta_data	
		pam_unix_authentication (authentication)	
	☐ name	session opened	
	☐ process	sudo	
	☐ signature	session opened for user	
	☐ src	splunk	
	☐ src_user	splunk	
	☐ src_user_id	1000	
	☐ src_user_name	splunk	
	☐ src_user_type	user	
	☐ tag	authentication	
		local	
		os	
		privileged	
		unix	
	☐ uid	0	
☐ user	root		
☐ user_id	0		
☐ user_name	root		
☐ user_role	administrator		
☐ user_type	user		
☐ vendor_product	nix		
Time	_time	2025-07-12T20:16:11.000+00:00	
Default	linecount	1	
	punct	"_":{};__= _	
	splunk_server	splunk	

Example Event Fields

Here are some of the fields you'll see in the search results:

- host: The name of your Linux server
- index: test (as configured)
- source: /var/log/auth.log
- sourcetype: linux_secure
- action: e.g., success for successful logins
- user: The username involved in the login
- _time: When the event happened

Onboarding Authentication Logs from Ubuntu UF

You can collect logs from /var/log/auth.log using two methods:

1. Using the full Splunk TA-linux_secure add-on
2. Using only inputs.conf for a lightweight setup

Prerequisites

- Splunk Universal Forwarder installed on Ubuntu
- SSH or console access to the server
- A running Splunk indexer
- A working outputs.conf file

1. Set Up outputs.conf

This file tells the Universal Forwarder where to send data.

Steps:

- Copy your existing outputs.conf to:
/home/ubuntu/splunkforwarder/etc/system/local/
- Use scp from your local machine:

```
C:\Windows\system32>scp "C:\Program Files\SplunkUniversalForwarder\etc\system\local\outputs.conf" ubuntu@192.168.1.15:/home/ubuntu/splunkforwarder/etc/system/local/  
ubuntu@192.168.1.15's password:  
outputs.conf 100% 152 0.2KB/s 00:00
```

```
ubuntu@ubuntu:~/splunkforwarder/etc/system/local$ cat outputs.conf  
[tcpout]  
defaultGroup = default-autolb-group  
  
[tcpout:default-autolb-group]  
server = 192.168.1.13:9997  
  
[tcpout-server://192.168.1.13:9997]
```

- Make sure the IP and port in outputs.conf match your Splunk indexer (default port is 9997)

2. Ensure Splunk User Can Read Log Files

The Splunk Forwarder runs under a specific user (often splunk). This user needs read access to /var/log/auth.log.

Add Read Permission

```
ubuntu@ubuntu:~$ sudo chmod o+r /var/log/auth.log
[sudo] password for ubuntu:
```

- This is less secure if the file contains sensitive data.

Check

```
ubuntu@ubuntu:~$ ls -l /var/log/auth.log
-rw-r--r-- 1 syslog adm 3594 Jul 13 15:49 /var/log/auth.log
```

- This is safer and more standard for log access.

3. Restarting the Universal Forwarder

After making changes (permissions, config files), restart the service:

```
ubuntu@ubuntu:~$ cd /home/ubuntu/splunkforwarder/bin
ubuntu@ubuntu:~/splunkforwarder/bin$ sudo ./splunk restart
[sudo] password for ubuntu:
Warning: Attempting to revert the SPLUNK_HOME ownership
Warning: Executing "chown -R ubuntu:ubuntu /home/ubuntu/splunkforwarder"
Stopping splunkd...
Shutting down. Please wait, as this may take a few minutes.

Stopping splunk helpers...

Done.
splunkd.pid doesn't exist...

Splunk> Be an IT superhero. Go home early.
```

4. Final Verification in Splunk

- Go to Splunk Web
- Search:

The screenshot shows the Splunk Web interface with a search query `index=test sourcetype=linux_secure` entered in the search bar. The search results show 67 events. The first four events are displayed in a table:

Time	Event
7/13/25 4:06:19.000 PM	Jul 13 16:06:19 ubuntu sudo: pam_unix(sudo:session): session closed for user root
7/13/25 4:06:09.000 PM	Jul 13 16:06:09 ubuntu sudo: pam_unix(sudo:session): session opened for user root(uid=0) by ubuntu(uid=1000)
7/13/25 4:06:09.000 PM	Jul 13 16:06:09 ubuntu sudo: ubuntu : TTY=ttty1 ; PWD=/home/ubuntu/splunkforwarder/bin ; USER=root ; COMMAND=./splunk restart
7/13/25 3:49:24.000 PM	Jul 13 15:49:24 ubuntu sudo: pam_unix(sudo:session): session closed for user root

- You should see parsed events from `/var/log/auth.log`, confirming successful onboarding!

Configuring Ingestion for Apache Web Server Access Logs

If you're running an Apache web server and want to send access logs to Splunk in a clean, structured way, here's a simple guide to help you do that using JSON format and the Splunk Universal Forwarder.

1. Configure Apache to Log in JSON Format

Splunk works best with structured data like JSON. So first, we'll set up Apache to log access data in JSON.

1.1 Define Custom Log Format

- Go to Apache config folder:

```
ubuntu@apache:~$ cd /etc/apache2/
ubuntu@apache:/etc/apache2$ ls -al
total 88
drwxr-xr-x  8 root root  4096 Jul 13 20:45 .
drwxr-xr-x 97 root root  4096 Jul 13 20:46 ..
-rw-r--r--  1 root root  7224 Apr  3 09:05 apache2.conf
drwxr-xr-x  2 root root  4096 Jul 13 20:45 conf-available
drwxr-xr-x  2 root root  4096 Jul 13 20:45 conf-enabled
-rw-r--r--  1 root root  1782 Mar 18  2024 envvars
-rw-r--r--  1 root root 31063 Mar 18  2024 magic
drwxr-xr-x  2 root root 12288 Jul 13 20:45 mods-available
drwxr-xr-x  2 root root  4096 Jul 13 20:45 mods-enabled
-rw-r--r--  1 root root   320 Mar 18  2024 ports.conf
drwxr-xr-x  2 root root  4096 Jul 13 20:45 sites-available
drwxr-xr-x  2 root root  4096 Jul 13 20:45 sites-enabled
```

- Create a new config file (e.g. log-splunk.conf) inside conf-available:

```
ubuntu@apache:/etc/apache2$ sudo nano conf-available/log-splunk.conf
```

- Add this custom log format:

```
GNU nano 6.2 conf-available/log-splunk.conf *
<IfModule log_config_module>
#
# The following directives define some format nicknames for use with
# a CustomLog directive (see below).
#
LogFormat "%h %l %u %t \"%r\" %>s %b \"%{Referer}i\" \"%{User-Agent}i\"" combined
LogFormat "%h %l %u %t \"%r\" %>s %b" common

<IfModule logio_module>
# You need to enable mod_logio.c to use %I and %O
<client=%h, status=%>s, uri_path=\"%U\", uri_query=\"%q\", user=\"%u\" splunk_json

#LogFormat "{\"time\":\"%{s}t.%{usec_frac}t\", \"bytes_in\":\"%I\", \"bytes_out\":\"%O\", \"cookie>

#LogFormat "%h %l %u %t \"%r\" %>s %b \"%{Referer}i\" \"%{User-Agent}i\" %I %O" combinedio

</IfModule>
#
# The location and format of the access logfile (Common Logfile Format).
# If you do not define any access logfiles within a <VirtualHost>
# container, they will be logged here. Contrariwise, if you *do*
# define per-<VirtualHost> access logfiles, transactions will be
# logged therein and *not* in this file.
#
# CustomLog "logs/access_log" common
#
# If you prefer a logfile with access, agent, and referer information
# (Combined Logfile Format) you can use the following directive.
#
CustomLog "/var/log/apache2/access.log" splunk_json
#CustomLog "logs/access_log" splunk_json
#CustomLog "logs/access_log" combined
</IfModule>
```

- Save and exit (Ctrl+X, then Y, then Enter).

1.2 Apply the JSON Format to Your Virtual Host

- Edit your default virtual host file:

```
ubuntu@apache:/etc/apache2$ sudo nano sites-available/000-default.conf
```

- Inside <VirtualHost *:80>, find or add this line:

```
GNU nano 6.2 sites-available/000-default.conf *
<VirtualHost *:80>
    # The ServerName directive sets the request scheme, hostname and port that
    # the server uses to identify itself. This is used when creating
    # redirection URLs. In the context of virtual hosts, the ServerName
    # specifies what hostname must appear in the request's Host: header to
    # match this virtual host. For the default virtual host (this file) this
    # value is not decisive as it is used as a last resort host regardless.
    # However, you must set it for any further virtual host explicitly.
    #ServerName www.example.com

    ServerAdmin webmaster@localhost
    DocumentRoot /var/www/html

    # Available loglevels: trace8, ..., trace1, debug, info, notice, warn,
    # error, crit, alert, emerg.
    # It is also possible to configure the loglevel for particular
    # modules, e.g.
    #LogLevel info ssl:warn

    ErrorLog ${APACHE_LOG_DIR}/error.log
    CustomLog ${APACHE_LOG_DIR}/access.log splunk_json

    # For most configuration files from conf-available/, which are
    # enabled or disabled at a global level, it is possible to
    # include a line for only one particular virtual host. For example the
    # following line enables the CGI configuration for this host only
    # after it has been globally disabled with "a2disconf".
    #Include conf-available/serve-cgi-bin.conf
</VirtualHost>

# vim: syntax=apache ts=4 sw=4 sts=4 sr noet
```

- Save and exit.
- Enable the config:

```
ubuntu@apache:/etc/apache2$ sudo a2enconf log-splunk
Enabling conf log-splunk.
To activate the new configuration, you need to run:
systemctl reload apache2
```

- Apply Apache Changes

```
ubuntu@apache:/etc/apache2/conf-enabled$ sudo systemctl reload apache2
```

1.3 Verify Apache is Logging in JSON

- Generate some traffic (e.g., open your website).

```
ubuntu@apache:/etc/apache2/conf-enabled$ curl http://127.0.0.1
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-trans
sitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
  <!--
    Modified from the Debian original for Ubuntu
    Last updated: 2022-03-22
    See: https://launchpad.net/bugs/1966004
  -->
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
    <title>Apache2 Ubuntu Default Page: It works</title>
    <style type="text/css" media="screen">
      *
      {
        margin: 0px 0px 0px 0px;
        padding: 0px 0px 0px 0px;
      }

      body, html {
        padding: 3px 3px 3px 3px;

        background-color: #D8DBE2;

        font-family: Ubuntu, Verdana, sans-serif;
        font-size: 11pt;
        text-align: center;
      }

      div.main_page {
        position: relative;
        display: table;

```

- Check the log:

```
ubuntu@apache:/var/log/apache2$ tail access.log
time=1752507052.017982, bytes_in=73, bytes_out=10926, cookie="-", server=127.0.1.1, dest_port=80, http_cont
ent_type="-", http_method="GET", http_referrer="-", http_user_agent="curl/7.81.0", ident="-", response_time
_microseconds=2580, client=127.0.0.1, status=200, uri_path="/index.html", uri_query="", user="-"
```

You should see JSON-formatted entries.

2. Install Splunk Universal Forwarder

- Go to Splunk Downloads and get the .tgz for Linux.

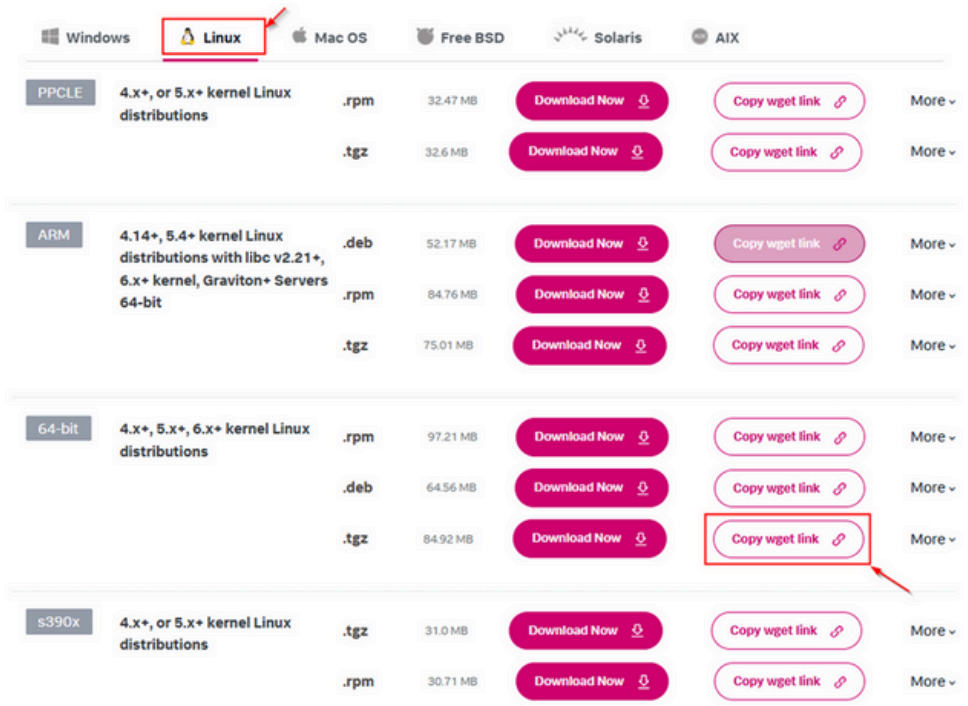
Universal Forwarder

The universal forwarder (UF) collects data securely from remote sources, including other forwarders, and sends it into Splunk software for indexing and consolidation. It's the primary way to send data into your Splunk Cloud Platform or Splunk Enterprise instance.

[Get My Free Download](#)



Getting External Data into Splunk



- Use wget to download it on your server.

```
ubuntu@apache:~$ wget -O splunkforwarder-9.4.3-237ebbd22314-linux-amd64.tgz "https://download.splunk.com/products/universalforwarder/releases/9.4.3/linux/splunkforwarder-9.4.3-237ebbd22314-linux-amd64.tgz"
--2025-07-14 16:01:23-- https://download.splunk.com/products/universalforwarder/releases/9.4.3/linux/splunkforwarder-9.4.3-237ebbd22314-linux-amd64.tgz
Resolving download.splunk.com (download.splunk.com)... 52.84.66.28, 52.84.66.112, 52.84.66.10, ...
Connecting to download.splunk.com (download.splunk.com)|52.84.66.28|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 89045711 (85M) [binary/octet-stream]
Saving to: 'splunkforwarder-9.4.3-237ebbd22314-linux-amd64.tgz'

splunkforwarder-9.4.3-2 100%[=====>] 84.92M 1.95MB/s in 43s

2025-07-14 16:02:11 (1.98 MB/s) - 'splunkforwarder-9.4.3-237ebbd22314-linux-amd64.tgz' saved [89045711/89045711]
```

- Extract it:

```
ubuntu@apache:~$ tar -xzf ./splunkforwarder-9.4.3-237ebbd22314-linux-amd64.tgz
splunkforwarder/
splunkforwarder/swidtag/
splunkforwarder/swidtag/splunk-UniversalForwarder-primary.swidtag
splunkforwarder/opt/
splunkforwarder/opt/jemalloc-4k-stats/
splunkforwarder/opt/jemalloc-4k-stats/include/
splunkforwarder/opt/jemalloc-4k-stats/include/jemalloc/
splunkforwarder/opt/jemalloc-4k-stats/include/jemalloc/jemalloc.h
splunkforwarder/opt/jemalloc-4k-stats/lib/
```

- Enable the forwarder:

```
ubuntu@apache:~$ sudo /home/ubuntu/splunkforwarder/bin/splunk enable boot-start -user ubuntu -systemd-managed 1
```

- Start the forwarder:

```
ubuntu@apache:~$ ./splunkforwarder/bin/splunk start
Warning: Attempting to revert the SPLUNK_HOME ownership
Warning: Executing "chown -R ubuntu:ubuntu /home/ubuntu/splunkforwarder"
Couldn't change ownership for /home/ubuntu/splunkforwarder/etc: Operation not permitted

Splunk> Needle. Haystack. Found.

Checking prerequisites...
  Checking mgmt port [8089]: open
  Checking conf files for problems...
  Done
  Checking default conf files for edits...
  Validating installed files against hashes from '/home/ubuntu/splunkforwarder/splunkforwarder-9.4.3-237ebb
d22314-linux-amd64-manifest'
  All installed files intact.
  Done
All preliminary checks passed.

Starting splunk server daemon (splunkd)...
==== AUTHENTICATING FOR org.freedesktop.systemd1.manage-units ====
Authentication is required to start 'SplunkForwarder.service'.
Authenticating as: ubuntu
Password:
==== AUTHENTICATION COMPLETE ====
Done
```

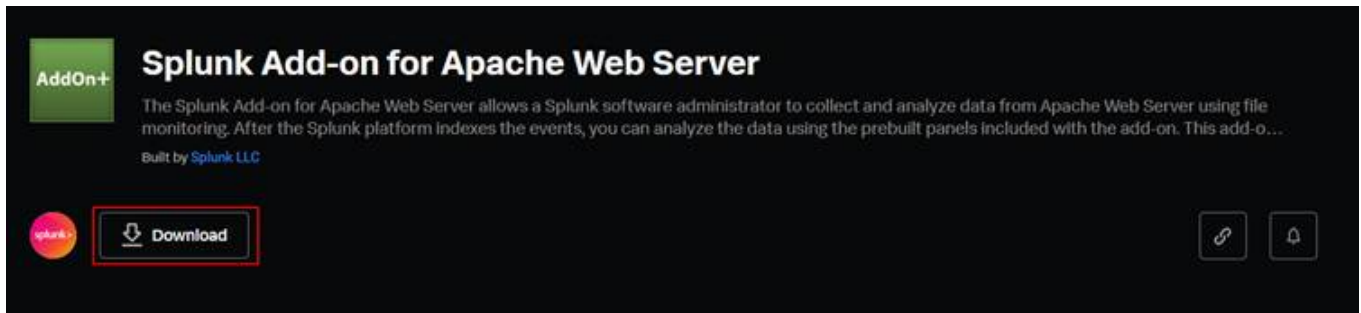
3. Install Splunk Add-on for Apache

Search for the appropriate Apache-related apps or add-ons on Splunkbase to enable log collection and analysis from Apache Web Servers within Splunk.

The screenshot shows the 'Browse More Apps' interface on Splunkbase. A search bar at the top left contains the text 'apache'. Below the search bar is a list of categories with checkboxes, including IT Operations, Security, Fraud & Compliance, Business Analytics, Utilities, Artificial Intelligence, IoT & Industrial Data, DevOps, Directory Service, Email, Endpoint, Firewall, Generic, Identity Management, Information, Investigative, Network Access Control, Network Device, Network Security, Reputation, Sandbox, and SIEM. The main content area displays search results for 'apache', showing 9 apps. The first app is 'Atlas ITSI Content Pack for Apache Web Server' by Presidio Splunk Solutions, with an 'Install' button. The second app is 'Splunk Add-on for Apache Web Server' by Splunk LLC, also with an 'Install' button. The interface includes filters for 'Best Match', 'Newest', and 'Popular'.

4. Download Splunk Add-on for Apache

- Go to Splunkbase and download the Splunk Add-on for Apache Web Server.



- Move it to the forwarder's etc/apps folder:

scp from local machine:

```
PS C:\WINDOWS\system32> scp C:\Users\pc\Downloads\splunk-add-on-for-apache-web-server_210.tgz ubuntu@192.168.1.17:/home/ubuntu
ubuntu@192.168.1.17's password:
splunk-add-on-for-apache-web-server_210.tgz 100% 37KB 12.0MB/s 00:00
```

mv on the server:

```
ubuntu@apache:~$ mv splunk-add-on-for-apache-web-server_210.tgz /home/ubuntu/splunkforwarder/etc/apps/
```

- Extract the add-on:

```
ubuntu@apache:~/splunkforwarder/etc/apps$ tar -xf splunk-add-on-for-apache-web-server_210.tgz
```

This will create a directory like Splunk_TA_apache.

- Restart the forwarder:

```
ubuntu@apache:~/splunkforwarder/bin$ sudo ./splunk restart
Warning: Attempting to revert the SPLUNK_HOME ownership
Warning: Executing "chown -R ubuntu:ubuntu /home/ubuntu/splunkforwarder"
Stopping splunkd...
Shutting down. Please wait, as this may take a few minutes.

Stopping splunk helpers...

Done.
splunkd.pid doesn't exist...

Splunk> Needle. Haystack. Found.

Checking prerequisites...
  Checking mgmt port [8089]: open
  Checking conf files for problems...
  Done
  Checking default conf files for edits...
  Validating installed files against hashes from '/home/ubuntu/splunkforwarder/splunkforwarder-9.4.3-237ebb
d22314-linux-amd64-manifest'
  All installed files intact.
  Done
All preliminary checks passed.

Starting splunk server daemon (splunkd)...
Done
```

5. Configure Log Monitoring

- Create the local folder if it doesn't exist:

```
ubuntu@apache:~/splunkforwarder/etc/apps/Splunk_TA_apache$ mkdir local
```

- Create inputs.conf:

```
ubuntu@apache:~/splunkforwarder/etc/apps/Splunk_TA_apache/local$ sudo nano inputs.conf
```

- Add this config:

```
GNU nano 6.2                               inputs.conf
[monitor:///var/log/apache2/access.log]
sourcetype = apache:access:json
index = test
disabled = 0
```

- Save and restart:

```
ubuntu@apache:~/splunkforwarder/bin$ sudo ./splunk restart
[sudo] password for ubuntu:
Warning: Attempting to revert the SPLUNK_HOME ownership
Warning: Executing "chown -R ubuntu:ubuntu /home/ubuntu/splunkforwarder"
Stopping splunkd...
Shutting down. Please wait, as this may take a few minutes.

Stopping splunk helpers...

Done.
splunkd.pid doesn't exist...

Splunk> Needle. Haystack. Found.

Checking prerequisites...
  Checking mgmt port [8089]: open
  Checking conf files for problems...
  Done
  Checking default conf files for edits...
  Validating installed files against hashes from '/home/ubuntu/splunkforwarder/splunkforwarder-9.4.3-237ebb
d22314-linux-amd64-manifest'
  All installed files intact.
  Done
All preliminary checks passed.

Starting splunk server daemon (splunkd)...
Done
```

6. Verify Logs in Splunk

- Go to Splunk Web.
- Open Search & Reporting.
- Run:

Getting External Data into Splunk

The screenshot shows the Splunk search interface. At the top, the search bar contains the query: `index=test host=apache sourcetype=apache:access:json`. Below the search bar, the search results are displayed in a table format. The table has two columns: **Time** and **Event**. The **Time** column shows the date and time of the event: 7/16/25 2:42:49.670 PM. The **Event** column shows the raw log data in JSON format, which is a detailed HTTP access log entry. The JSON data includes fields such as `bytes_in`, `bytes_out`, `client`, `cookie`, `dest_port`, `http_content_type`, `http_method`, `http_referer`, `http_user_agent`, `ident`, `response_time_microseconds`, `server`, `status`, `time`, `uri_path`, `uri_query`, and `user`. The interface also shows a sidebar on the left with various filters and a top bar with search and navigation options.

Time	Event
7/16/25 2:42:49.670 PM	<pre>{ "bytes_in": 73, "bytes_out": 18926, "client": "127.0.0.1", "cookie": "", "dest_port": 80, "http_content_type": "", "http_method": "GET", "http_referer": "", "http_user_agent": "curl/7.81.0", "ident": "", "response_time_microseconds": 324, "server": "127.0.0.1", "status": 200, "time": 1752679369.670181, "uri_path": "/index.html", "uri_query": "", "user": "" }</pre>

You should see your Apache logs in JSON format

Comprehensive Guide to Onboarding CSV Files

CSV files are everywhere simple, structured, and easy to work with. But when it comes to Splunk, you need to make sure the data is ingested properly so you can search and analyze it easily. Here's a step-by-step guide to help you do that using Indexed Extractions.

What Are Indexed Extractions?

Normally, Splunk reads each line of a file as a raw event. That means even the header (like rank,domain,tld) shows up as an event.

With Indexed Extractions, Splunk reads the header and uses it to extract fields from each line. So instead of raw lines, you get structured events like:

rank=1, domain=google.com, tld=com

```
ubuntu@ubuntu:~$ ls -la /var/log/top*
-rw-rw-r-- 1 ubuntu ubuntu 28941 Jul 17 13:00 /var/log/top-1000.csv
-rw-rw-r-- 1 ubuntu ubuntu 28941 Jul 18 10:45 /var/log/top-1000.txt
ubuntu@ubuntu:~$ head /var/log/top-1000.csv
# These are the top 1000 domains and their top level domains
rank,domain,tld
1,google.com,com
2,microsoft.com,com
3,www.google.com,com
4,netflix.com,com
5,cloud.netflix.com,com
6,prod.cloud.netflix.com,com
7,data.microsoft.com,com
8,ftl.netflix.com,com
```

- **Important:** All parsing and field extraction must happen on the Universal Forwarder. Settings on the Indexer won't work here.

Monitor Your CSV File

To tell Splunk to watch your CSV file for new data:

- Create top1000_csv_inputs and default folders
- Go to your app's inputs directory
- Create inputs.conf:

Getting External Data into Splunk

```
ubuntu@ubuntu:~$ cd ./splunkforwarder/etc/apps/  
ubuntu@ubuntu:~/splunkforwarder/etc/apps$ mkdir top1000_csv_inputs  
ubuntu@ubuntu:~/splunkforwarder/etc/apps$ cd top1000_csv_inputs/  
ubuntu@ubuntu:~/splunkforwarder/etc/apps/top1000_csv_inputs$ mkdir default  
ubuntu@ubuntu:~/splunkforwarder/etc/apps/top1000_csv_inputs$ cd default/  
ubuntu@ubuntu:~/splunkforwarder/etc/apps/top1000_csv_inputs/default$ nano inputs.conf
```

- Add this stanza:

```
GNU nano 6.2 inputs.conf *  
[monitor:///var/log/top-1000.csv]  
index = test  
disabled = 0
```

Verify Ingestion in Splunk

- Go to Splunk Web and search:

New Search

index=test source="/var/log/top-1000.csv"

1,000 events (7/17/25 12:00:00:000 AM to 7/17/25 10:32:00:000 PM) No Event Sampling

Events (1,000) Patterns Statistics Visualization

Timeline format Zoom Out Zoom to Selection Deselect

Format Show: 20 Per Page View: List

i	Time	Event
>	7/17/25 100:19:00 PM	1000,pull-his-f77-sg01.tiktokcdn.com,com
>	7/17/25 100:19:00 PM	999,syndication.twitter.com,com
>	7/17/25 100:19:00 PM	998,pull-as-sg01.tiktokcdn.com,com
>	7/17/25 100:19:00 PM	997,r1b.afadsrvr.com,com
>	7/17/25 100:19:00 PM	996,pull-his-f58-sg01.tiktokcdn.com,com
>	7/17/25 100:19:00 PM	995,pull-cmaf-f16-tt02.tiktokcdn.com,com

Selected Fields: host, index, source, sourcetype

- Click on an event and check the Selected Fields. You should see fields like rank, domain, and tld.

Selected Fields

tld

15 Values, 99.8% of events

Selected Yes No

Reports

Top values Top values by time Rare values

Events with this field

Top 10 Values	Count	%
com	765	76.653%
net	175	17.535%
io	19	1.904%
org	16	1.603%
ms	5	0.501%
apple	3	0.301%
fi	2	0.2%
goog	2	0.2%
internal	2	0.2%
me	2	0.2%

Selected Fields: host, index, source, sourcetype

Interesting Fields: domain, eventtype, linecount, rank, splunk_server, timestamp

Getting External Data into Splunk

New Search

index=test source="/var/log/top-1000.csv" *

19 events (7/17/25 12:00:00.000 AM to 7/17/25 12:48:00.000 PM) No Event Sampling

Events (19) Patterns Statistics Visualization

Timeline format Zoom Out Zoom to Selection Deselect

Format Show 20 Per Page View List

Time	Event
7/17/25 10:01:19.000 PM	981,segment.io,io host = ubuntu index = test source = /var/log/top-1000.csv sourcetype = csv
7/17/25 10:01:19.000 PM	955,cdn.linkedin.oribl.io,io host = ubuntu index = test source = /var/log/top-1000.csv sourcetype = csv
7/17/25 10:01:19.000 PM	951,linkedin.oribl.io,io host = ubuntu index = test source = /var/log/top-1000.csv sourcetype = csv
7/17/25 10:01:19.000 PM	938,oribl.io,io host = ubuntu index = test source = /var/log/top-1000.csv sourcetype = csv
7/17/25 10:01:19.000 PM	936,notify.adobe.io,io host = ubuntu index = test source = /var/log/top-1000.csv sourcetype = csv
7/17/25 10:01:19.000 PM	938,data.pendo.io,io host = ubuntu index = test source = /var/log/top-1000.csv sourcetype = csv

Improve Field Extraction (props.conf)

If Splunk doesn't extract fields correctly, you can force it to treat the file as CSV:

- Go to: `~/splunkforwarder/etc/system/default/`
- Edit `props.conf`:

```
ubuntu@ubuntu:~/splunkforwarder/etc/system/default$ nano props.conf
```

- Add this:

```
[csv]
SHOULD_LINEMERGE = False
pulldown_type = true
INDEXED_EXTRactions = csv
KV_MODE = none
category = Structured
description = Comma-separated value format. Set header and other settings in "Delimited Settings"
```

- Check:

Hide Fields All Fields

SELECTED FIELDS

- host 1
- index 1
- source 1
- sourcetype 1

INTERESTING FIELDS

- domain 100+
- eventtype 2
- linecount 1
- rank 100+
- splunk_server 1
- timestamp 1
- tid 15

+ Extract New Fields

sourcetype

1 Value, 100% of events

Selected Yes No

Reports

- Top values
- Top values by time
- Rare values

Events with this field

Values	Count	%
CSV	1,000	100%

7/17/25 1:00:19.000 PM 996,pull-hls-f58-sg01.tiktokcdn.com,com host = ubuntu index = test source = /var/log/top-1000.csv sourcetype = csv

7/17/25 1:00:19.000 PM 995,pull-cmaf-f16-tt02.tiktokcdn.com,com host = ubuntu index = test source = /var/log/top-1000.csv sourcetype = csv

7/17/25 994,pull-cmaf-l10-sg01.tiktokcdn.com,com

Handle Multiple Files with Different Sourcetypes

If you have more than one file (e.g., .txt or other formats), give each one a unique sourcetype:

```
ubuntu@ubuntu:~/splunkforwarder/etc/system/default$ head /var/log/top-1000.txt
# These are the top 1000 domains and their top level domains
rank, domain, tld
1, google.com, com
2, microsoft.com, com
3, www.google.com, com
4, netflix.com, com
5, cloud.netflix.com, com
6, prod.cloud.netflix.com, com
7, data.microsoft.com, com
8, ftl.netflix.com, com
```

- Edit inputs.conf again:

```
ubuntu@ubuntu:~/splunkforwarder/etc/apps/top1000_csv_inputs/default$ nano ./inputs.conf
```

```
GNU nano 6.2                               ./inputs.conf *
[monitor:///var/log/top-1000.csv]
index = test
disabled = 0

[monitor:///var/log/top-1000.txt]
sourcetype = top_domains
index = test
crcSalt = splunk4analysts
disabled = 0
```

This helps you search and manage data more easily.

Restart Splunk and Confirm

- Restart the forwarder:

```
ubuntu@ubuntu:~/splunkforwarder$ sudo ./bin/splunk restart
[sudo] password for ubuntu:
Warning: Attempting to revert the SPLUNK_HOME ownership
Warning: Executing "chown -R ubuntu:ubuntu /home/ubuntu/splunkforwarder"
Stopping splunkd...
Shutting down. Please wait, as this may take a few minutes.

Stopping splunk helpers...
Done.
```

Getting External Data into Splunk

- Go back to Splunk Web and search:

The screenshot shows the Splunk Web interface with a search for `index=test`. The search results show 2,071 events. A modal window titled "sourcetype" is open, displaying a bar chart and a table of values. The table shows the following data:

Values	Count	%
csv	1,000	48.28%
top_domains	1,000	48.28%
linux_secure	71	3.42%

The screenshot shows the Splunk Web interface with a search for `index=test sourcetype=top_domains`. The search results show 1,000 events. The table below displays the first few events:

Time	Event
7/18/25 10:45:21.000 AM	1000,pull-his-f77-sg81.tiktokcdn.com,com
7/18/25 10:45:21.000 AM	999,syndication.twitter.com,com
7/18/25 10:45:21.000 AM	998,pull-v5-sg81.tiktokcdn.com,com
7/18/25 10:45:21.000 AM	997,rtb.afadrivr.com,com
7/18/25 10:45:21.000 AM	996,pull-his-f58-sg81.tiktokcdn.com,com
7/18/25 10:45:21.000 AM	995,pull-cmaf-f18-t182.tiktokcdn.com,com
7/18/25 10:45:21.000 AM	994,pull-cmaf-118-sg81.tiktokcdn.com,com

- Check if fields like tld are extracted correctly. You can even explore top values to confirm data quality.

The screenshot shows the Splunk Web interface with a search for `index=test sourcetype=top_domains`. The search results show 1,000 events. A modal window titled "tld" is open, displaying a bar chart and a table of top values. The table shows the following data:

Top 10 Values	Count	%
com	765	76.653%
net	175	17.535%
io	19	1.904%
org	16	1.603%
ms	5	0.501%
apple	3	0.301%
fi	2	0.2%
goog	2	0.2%
internal	2	0.2%
me	2	0.2%

Custom Data Sources and the "Great 8" Best Practices

When you're working with custom logs, like niche software, in-house apps, or unique formats, Splunk won't have a ready-made add-on. That's where custom sourcetypes and Technical Add-ons (TAs) come in.

To make sure Splunk parses your data correctly, you need to configure how events are broken, timestamped, and indexed. This is done using the "Great 8" parsing settings in props.conf.

The "Great 8" Parsing Settings Explained

Here are the 8 key settings you should always define when creating a custom sourcetype:

1. **TIME_PREFIX**

- Tells Splunk where the timestamp starts.
- Example: timestamp=" if your log looks like timestamp="2023-10-26..."

2. **TIME_FORMAT**

- Defines the format of the timestamp.
- Example: %Y-%m-%d %H:%M:%S for 2023-10-26 14:30:00

3. **MAX_TIMESTAMP_LOOKAHEAD**

- How many characters Splunk should scan after TIME_PREFIX to find the timestamp.

4. **SHOULD_LINEMERGE**

- Set to false if each log entry is one line. Helps performance.

5. **LINE_BREAKER**

- Regex that defines where one event ends and the next begins.
- Example: ([\r\n]+) for new lines.

6. **TRUNCATE**

- Max size of a single event (in bytes). Prevents huge events from being indexed.

7. **EVENT_BREAKER**

- Lets Splunk break events at the Universal Forwarder level (not just indexer).

8. **EVENT_BREAKER_ENABLE**

- Enables the above setting. Set to true.

Example: Onboarding Ubuntu dpkg.log

Let's say you want to ingest /var/log/dpkg.log from an Ubuntu system.

1. Create App Directory

```
ubuntu@ubuntu:~/splunkforwarder/etc/apps$ mkdir -p dpkg/default
ubuntu@ubuntu:~/splunkforwarder/etc/apps$ cd dpkg/default/
```

2. Define Parsing Rules in props.conf

```
ubuntu@ubuntu:~/splunkforwarder/etc/apps/dpkg/default$ nano props.conf
```

- **props.conf Content for dpkg Logs**

```
GNU nano 6.2                                props.conf *
[ubuntu_dpkg_log]
TIME_PREFIX = ^
TIME_FORMAT = %Y-%m-%d %H:%M:%S
MAX_TIMESTAMP_LOOKAHEAD = 20
SHOULD_LINEMERGE = false
LINE_BREAKER = ([\r\n]+)
TRUNCATE = 2000
EVENT_BREAKER = ([\r\n]+)
EVENT_BREAKER_ENABLE = true
```

This configuration explicitly sets several key parsing parameters for the `ubuntu_dpkg_log` sourcetype:

- **TIME_PREFIX = ^**: Indicates the timestamp is expected at the beginning of each event.
- **TIME_FORMAT = %Y-%m-%d %H:%M:%S**: Specifies the timestamp format (e.g., 2023-10-26 14:30:00).
- **MAX_TIMESTAMP_LOOKAHEAD = 20**: Sets the maximum characters Splunk will scan for the timestamp after the `TIME_PREFIX`.
- **SHOULD_LINEMERGE = false**: Prevents multiple lines from being combined into a single event, indicating each log entry is a single line.
- **LINE_BREAKER = ([\r\n]+)**: Defines that a new line character signifies a new event, consistent with `SHOULD_LINEMERGE = false`.
- **TRUNCATE = 2000**: Sets the maximum event size to 2000 bytes, acting as a safeguard against overly large events.
- **EVENT_BREAKER = ([\r\n]+)**: Enables event breaking at the Universal Forwarder level based on new lines.
- **EVENT_BREAKER_ENABLE = true**: Activates the `EVENT_BREAKER` functionality on the forwarder.

3. Configure Input in inputs.conf

```
ubuntu@ubuntu:~/splunkforwarder/etc/apps/dpkg/default$ nano inputs.conf
```

```
GNU nano 6.2                               inputs.conf *
[monitor:///var/log/dpkg.log]
sourcetype = dpkg
index = test
disabled = 0
```

This configuration specifies:

- **[monitor:///var/log/dpkg.log]:** Instructs Splunk to monitor the file /var/log/dpkg.log.
- **sourcetype = dpkg:** Assigns the dpkg sourcetype to the data from this log file, ensuring it uses the parsing rules defined in props.conf.
- **index = test:** Directs the ingested data to the test index in Splunk.
- **disabled = 0:** Ensures this input is active (0 means enabled).

4. Restart the Forwarder

```
ubuntu@ubuntu:~/splunkforwarder$ sudo ./bin/splunk restart
[sudo] password for ubuntu:
Warning: Attempting to revert the SPLUNK_HOME ownership
Warning: Executing "chown -R ubuntu:ubuntu /home/ubuntu/splunkforwarder"
Stopping splunkd...
Shutting down. Please wait, as this may take a few minutes.

Stopping splunk helpers...

Done.
splunkd.pid doesn't exist...

Splunk> 4TW

Checking prerequisites...
  Checking mgmt port [8089]: open
  Checking conf files for problems...
  Done
  Checking default conf files for edits...
  Validating installed files against hashes from '/home/ubuntu/splunkforwarder/splunkforwarder-9.4.3-237ebbd22314-linux-amd64-manifest'
  All installed files intact.
  Done
All preliminary checks passed.

Starting splunk server daemon (splunkd)...
Done
```

5. Verify in Splunk Web

- Search:

The screenshot shows the Splunk Web search interface. The search bar contains the query `index=test sourcetype=dpkg`. Below the search bar, it indicates 145 events from 7/13/25 9:00:00.000 AM to 7/20/25 9:08:39.000 AM. The interface includes tabs for Events (145), Patterns, Statistics, and Visualization. The Events tab is active, showing a list of events with columns for Time and Event. The left sidebar shows fields like host, index, source, and sourcetype. The main table displays the following events:

Time	Event
7/20/25 9:08:22.000 AM	2025-07-20 09:08:22 status installed man-db:amd64 2.10.2-1 host = ubuntu index = test source = /var/log/dpkg.log sourcetype = dpkg
7/20/25 9:08:22.000 AM	2025-07-20 09:08:22 status half-configured man-db:amd64 2.10.2-1 host = ubuntu index = test source = /var/log/dpkg.log sourcetype = dpkg
7/20/25 9:08:22.000 AM	2025-07-20 09:08:22 trigger man-db:amd64 2.10.2-1 <none> host = ubuntu index = test source = /var/log/dpkg.log sourcetype = dpkg
7/20/25 9:08:22.000 AM	2025-07-20 09:08:22 status installed net-tools:amd64 1.60-g120181103.0eebece-ubuntu5.4 host = ubuntu index = test source = /var/log/dpkg.log sourcetype = dpkg
7/20/25 9:08:22.000 AM	2025-07-20 09:08:22 status half-configured net-tools:amd64 1.60-g120181103.0eebece-ubuntu5.4 host = ubuntu index = test source = /var/log/dpkg.log sourcetype = dpkg
7/20/25 9:08:22.000 AM	2025-07-20 09:08:22 status unpacked net-tools:amd64 1.60-g120181103.0eebece-ubuntu5.4 host = ubuntu index = test source = /var/log/dpkg.log sourcetype = dpkg
7/20/25 9:08:22.000 AM	2025-07-20 09:08:22 configure net-tools:amd64 1.60-g120181103.0eebece-ubuntu5.4 <none> host = ubuntu index = test source = /var/log/dpkg.log sourcetype = dpkg
7/20/25 9:08:22.000 AM	2025-07-20 09:08:22 startup packages configure host = ubuntu index = test source = /var/log/dpkg.log sourcetype = dpkg
7/20/25	2025-07-20 09:08:22 status unpacked net-tools:amd64 1.60-g120181103.0eebece-ubuntu5.4

You should see clean, timestamped events from dpkg.log, parsed using your custom sourcetype.

Extracting Data Fields using the EXTRACT Command

Splunk is powerful because it doesn't just store logs, it makes data searchable. One of the key ways it does this is through field extraction, which turns raw log lines into structured, searchable fields.

What Is Search-Time Field Extraction?

Unlike index-time extraction (which happens when data first enters Splunk), search-time extraction happens when you run a search. This means:

- You can define new fields anytime, even after data is already indexed.
- No need to re-ingest or re-index your data.
- Super flexible, perfect for evolving log formats or changing analysis needs.

Things to Keep in Mind

- Search-time extraction uses CPU on the Search Head every time you run a query.
- In large environments, make sure field extraction configs are consistent across all Search Heads, especially in clusters.

How to Extract Fields in Splunk

Splunk offers three main methods to extract fields:

1. EXTRACT (Inline in props.conf)

- Define regex directly in props.conf under your sourcetype.
- Best for simple to medium complexity extractions.

2. REPORT (Using transforms.conf)

- Use props.conf + transforms.conf for more complex logic or reusable regex.
- Great for shared rules across multiple sourcetypes or when using lookups.

3. Automatic Field Discovery

Splunk can auto-extract fields from structured formats like:

- Key-Value pairs: user=jsmith status=success
- JSON: { "user": "jsmith", "status": "success" }
- XML: <user>jsmith</user>

No config needed, Splunk does the work for you!

Step-by-Step: Extracting Fields via UI

```
splunk@splunk:~$ tail /var/log/fail2ban.log
2025-07-21 11:52:08,288 fail2ban.jail [69865]: INFO Creating new jail 'sshd'
2025-07-21 11:52:08,305 fail2ban.jail [69865]: INFO Jail 'sshd' uses pyinotify {}
2025-07-21 11:52:08,311 fail2ban.jail [69865]: INFO Initiated 'pyinotify' backend
2025-07-21 11:52:08,313 fail2ban.filter [69865]: INFO maxLines: 1
2025-07-21 11:52:08,340 fail2ban.filter [69865]: INFO maxRetry: 5
2025-07-21 11:52:08,340 fail2ban.filter [69865]: INFO findtime: 600
2025-07-21 11:52:08,340 fail2ban.actions [69865]: INFO banTime: 600
2025-07-21 11:52:08,341 fail2ban.filter [69865]: INFO encoding: UTF-8
2025-07-21 11:52:08,342 fail2ban.filter [69865]: INFO Added logfile: '/var/log/auth.log' (pos = 0, has
h = 4f91f2b22f109ae4b6c9210830dc14ed35fffa9f)
2025-07-21 11:52:08,372 fail2ban.jail [69865]: INFO Jail 'sshd' started
splunk@splunk:~$ cd ./splunk/etc/apps/fail2ban/default/
splunk@splunk:~/etc/apps/fail2ban/default$ cat props.conf
[fail2ban_logs]
TIME_PREFIX = ^
TIME_FORMAT = %Y-%m-%d %H:%M:%S,%N
MAX_TIMESTAMP_LOOKAHEAD = 24
SHOULD_LINEMERGE = false
LINE_BREAKER = ([\r\n]+)
TRUNCATE = 10000
EVENT_BREAKER = ([\r\n]+)
EVENT_BREAKER_ENABLE = true
```

1. Search your data

Example:

The screenshot shows the Splunk search interface. The search bar contains the query `index=test sourcetype=fail2ban`. Below the search bar, there are tabs for **Events (15)**, **Patterns**, **Statistics**, and **Visualization**. The **Events (15)** tab is selected, showing a list of search results. The results table has columns for **Time** and **Event**. The first event is from 7/21/25 at 11:52:08.372 AM, with the event text: `2025-07-21 11:52:08,372 fail2ban.jail [69865]: INFO Jail 'sshd' started`. The second event is from 7/21/25 at 11:52:08.342 AM, with the event text: `2025-07-21 11:52:08,342 fail2ban.filter [69865]: INFO Added logfile: '/var/log/auth.log' (pos = 0, hash = 4f91f2b22f109ae4b6c9210830dc14ed35fffa9f)`. The third event is from 7/21/25 at 11:52:08.341 AM, with the event text: `2025-07-21 11:52:08,341 fail2ban.filter [69865]: INFO encoding: UTF-8`. The fourth event is from 7/21/25 at 11:52:08.340 AM, with the event text: `2025-07-21 11:52:08,340 fail2ban.actions [69865]: INFO banTime: 600`. The fifth event is from 7/21/25 at 11:52:08.340 AM, with the event text: `2025-07-21 11:52:08,340 fail2ban.filter [69865]: INFO findtime: 600`. The sixth event is from 7/21/25 at 11:52:08.340 AM, with the event text: `2025-07-21 11:52:08,340 fail2ban.filter [69865]: INFO maxRetry: 5`.

2. Click “Event Actions” → “Extract Fields”

The screenshot shows the Splunk Event Actions menu. The menu is open, and the **Extract Fields** option is highlighted. The menu also includes **Build Event Type** and **Show Source**. Below the menu, there is a table with columns for **Value** and **Actions**. The table contains the following rows:

Value	Actions
splunk	▼
test	▼
source	▼
source	▼
eventtype	▼
eventtype	▼
_time	▼
_time	▼
linecount	▼
linecount	▼
punct	▼
punct	▼
splunk_server	▼
splunk_server	▼

3. Choose “Regular Expression” method

Extract Fields

Select Method Select Fields Validate Save **Next >** Existing fields >

Select Method

Indicate the method you want to use to extract your field(s). [Learn more](#)

I prefer to write the regular expression myself >

Source type
fail2ban

2025-07-21 11:52:08,372 fail2ban.jail [69865]: INFO Jail 'sshd' started

(.*?)
Regular Expression
Splunk Enterprise will extract fields using a Regular Expression.

x|y|z
Delimiters
Splunk Enterprise will extract fields using a delimiter (such as commas, spaces, or characters). Use this method for delimited data like comma separated values (CSV files).

4. Highlight the value you want to extract

Example: highlight INFO to create a field called log_level.

5. Name the field and click “Add Extraction”

Extract Fields

Select Method Select Fields Validate Save < Back **Next >** Existing fields >

Select Fields

Highlight one or more values in the sample event to create fields. You can indicate one value is required, meaning it must exist in an event for the regular expression to match. Click on highlighted values in the sample event to modify them. To highlight text that is already part of an existing extraction, first turn off the existing extractions. [Learn more](#)

2025-07-21 11:52:08,372 fail2ban.jail [69865]: **INFO** Jail 'sshd' started

1

Extract Require

Field Name **log_level** 2

Sample Value **INFO**

3 **Add Extraction**

6. Preview and refine the regex if needed

Preview

If you see incorrect results below, click an additional event to add it to the set of sample events. Highlight its values to improve the extraction. You can remove incorrect values in the next step.

Events ☒ log_level

✓ 15 events (4/22/25 12:00:00.000 AM to 7/21/25 1:08:06.000 PM) 20 per page ▾

filter

✓ raw 2	log_level 1
✓ 2025-07-21 11:52:08,372 fail2ban.jail [69865]: INFO Jail 'sshd' started	INFO
✓ 2025-07-21 11:52:08,342 fail2ban.filter [69865]: INFO Added logfile: '/var/log/auth.log' (pos = 0, hash = 4f91f2b22f189ae4b6c9210830dc14ed35fffa9f)	INFO
✓ 2025-07-21 11:52:08,341 fail2ban.filter [69865]: INFO encoding: UTF-8	INFO
✓ 2025-07-21 11:52:08,340 fail2ban.actions [69865]: INFO banTime: 600	INFO
✓ 2025-07-21 11:52:08,340 fail2ban.filter [69865]: INFO findTime: 600	INFO
✓ 2025-07-21 11:52:08,340 fail2ban.filter [69865]: INFO maxRetry: 5	INFO
✓ 2025-07-21 11:52:08,313 fail2ban.filter [69865]: INFO maxLines: 1	INFO
✓ 2025-07-21 11:52:08,311 fail2ban.jail [69865]: INFO Initiated 'pyinotify' backend	INFO
✓ 2025-07-21 11:52:08,305 fail2ban.jail [69865]: INFO Jail 'sshd' uses pyinotify {}	INFO
✓ 2025-07-21 11:52:08,288 fail2ban.jail [69865]: INFO Creating new jail 'sshd'	INFO
✓ 2025-07-21 11:52:08,288 fail2ban.database [69865]: WARNING New database created. Version '4'	WARNING
✓ 2025-07-21 11:52:08,288 fail2ban.database [69865]: INFO Connected to fail2ban persistent database '/var/lib/fail2ban/fail2ban.sqlite'	INFO
✓ 2025-07-21 11:52:08,267 fail2ban.observer [69865]: INFO Observer start...	INFO
✓ 2025-07-21 11:52:08,266 fail2ban.server [69865]: INFO Starting Fail2ban v0.11.2	INFO
✓ 2025-07-21 11:52:08,266 fail2ban.server [69865]: INFO -----	INFO

Extract Fields ☒ Select Method ☒ Select Fields ☐ Validate ☐ Save

Select Fields

Highlight one or more values in the sample event to create fields. You can indicate one value is required, meaning it must exist in an event for the regular expression to match. Click on highlighted values in the sample event to modify them. To highlight text that is already part of an existing extraction, first turn off the existing extractions. [Learn more](#)

2025-07-21 11:52:08,372 fail2ban.jail [69865]: INFO Jail 'sshd' started

Hide Regular Expression ▾

Preview

If you see incorrect results below, click an additional event to add it to the set of sample events. Highlight its values to improve the extraction. You can remove incorrect values in the next step.

Events ☒ log_level

✓ 15 events (4/22/25 12:00:00.000 AM to 7/21/25 1:08:06.000 PM) 20 per page ▾

7. Save the extraction and set permissions

Extract Fields

⚠ If you manually edit and then preview the regular expression below, you cannot return to the automatic field extraction workflow.

Use the event listing below to validate the field extractions produced by your regular expression.

Regular Expression [Regular Expression Reference 🔍](#) [View in Search 🔍](#)

Events ☒ log_level

✓ 15 events (4/22/25 12:00:00.000 AM to 7/21/25 1:08:06.000 PM) 20 per page ▾

filter

Getting External Data into Splunk

Extract Fields < Back Finish

Save
Name the extraction and set permissions.

Extractions Name:

Owner:

App:

Permissions:

	Owner	App	All apps
			Read Write
Everyone	☒	☒	☒
admin	☐	☒	☒
can_delete	☐	☐	☐
power	☐	☐	☐
splunk-system-role	☐	☐	☐
splunk_system_upgrader	☐	☐	☐
user	☐	☐	☐

Source type:

Fields:

Regular Expression:

8. Explore your new field in search results

Success!

You have extracted additional fields from your data (sourcetype=fail2ban).

Edit your field extractions at any time by going to [Field Extractions](#).

What would you like to do next?

→ [Explore the fields I just created in Search](#)

→ [Extract more fields](#)

New Search Save As Create Table View Close

Last 24 hours Q

✓ 17 events (7:20:25 1:00:00:00 PM to 7:29:25 1:59:00:00 PM) No Event Sampling

Events (17) Patterns Statistics Visualization

Timeline format Zoom Out Zoom to Selection Developt Hour per column

Format Show: 20 Per Page View List

Time	Event
7/21/25 12:51:59.312 PM	[69865]: WARNING [sshd] Please check jail has possibly a timezone issue. Line with odd timestamp: Jul 21 12:17:26 splunk mongod: looking for plugins in '/home/build/build-home/opt/mongo/lib/sasl2', failed to open directory, error: No such file or directory host = splunk index = test source = /var/log/fail2ban.log sourcetype = fail2ban
7/21/25 12:51:59.307 PM	[69865]: WARNING [sshd] Simulate NOW in operation since found time has too large deviation 1753188246.8 - 1753182319.385823 +/- 68 host = splunk index = test source = /var/log/fail2ban.log sourcetype = fail2ban
7/21/25 11:52:08.372 AM	[69865]: INFO Jail 'sshd' started host = splunk index = test source = /var/log/fail2ban.log sourcetype = fail2ban
7/21/25 11:52:08.342 AM	[69865]: INFO Added logfile: '/var/log/auth.log' (pos = 8, hash = 4f91f2b22f18ba4b6c9218838dc14ed35fffa9f) host = splunk index = test source = /var/log/fail2ban.log sourcetype = fail2ban
7/21/25 11:52:08.341 AM	[69865]: INFO encoding: UTF-8 host = splunk index = test source = /var/log/fail2ban.log sourcetype = fail2ban

7/21/25 12:51:59.312 PM [69865]: WARNING [sshd] Please check jail has possibly a timezone issue. Line with odd timestamp: Jul 21 12:17:26 splunk mongod: looking for plugins in '/home/build/build-home/opt/mongo/lib/sasl2', failed to open directory, error: No such file or directory

Event Actions

Type	Field	Value	Actions
Selected	host	splunk	
	index	test	
	source	/var/log/fail2ban.log	
	sourcetype	fail2ban	
Event	eventtype	nix-all-logs	
		nix_errors (error)	
		nix_ta_data	
	log_level	WARNING	
	tag	error	
Time	_time	2025-07-21T12:51:59.312+00:00	
Default	linecount	1	
	punct		
	splunk_server	splunk	

Extracting Data Fields using the REPORT Command

Why Use REPORT for Field Extraction in Splunk?

When working with custom log formats in Splunk, like **fail2ban logs** or other non-standard sources, you often need to extract fields using regular expressions (regex). Splunk gives you two main ways to do this: **EXTRACT** and **REPORT**.

While **EXTRACT** is great for quick setups, **REPORT** is the better choice when you want clean, reusable, and scalable configurations.

Benefits of Using REPORT

1. Regex Reusability

Define your regex once in `transforms.conf` and reuse it across multiple sourcetypes in `props.conf`.

2. Modular Configuration

Keeps your regex logic separate from sourcetype definitions — easier to manage and troubleshoot.

3. Advanced Field Mapping

Use **FORMAT** to assign names to regex capture groups, making your field extraction more readable and precise.

Example: Extracting Fields from fail2ban Logs

1. Define the Extraction in transforms.conf

```
splunk@splunk:~/splunk/etc/apps/fail2ban/default$ nano transforms.conf
```

```
GNU nano 6.2 transforms.conf
[fail2ban_fields]
REGEX = ^(<timestamp>\d{4}-\d{2}-\d{2}\s+\d{2}:\d{2}:\d{2},\d{3})
FORMAT = timestamp::$1 component::$2 pid::$3 level::$4 jail::$5 message::$6
```

- This regex captures key parts of the log and maps them to field names like `timestamp`.

2. Apply the Extraction in props.conf

```
splunk@splunk:~/splunk/etc/apps/fail2ban/default$ nano props.conf
```

```
GNU nano 6.2                                props.conf
[fail2ban]
SHOULD_LINEMERGE = false
REPORT-fail2ban = fail2ban_fields
TIME_PREFIX = ^
TIME_FORMAT = %Y-%m-%d %H:%M:%S,%3N
MAX_TIMESTAMP_LOOKAHEAD = 24
```

- This tells Splunk to apply the fail2ban_fields extraction to all events with sourcetype=fail2ban.

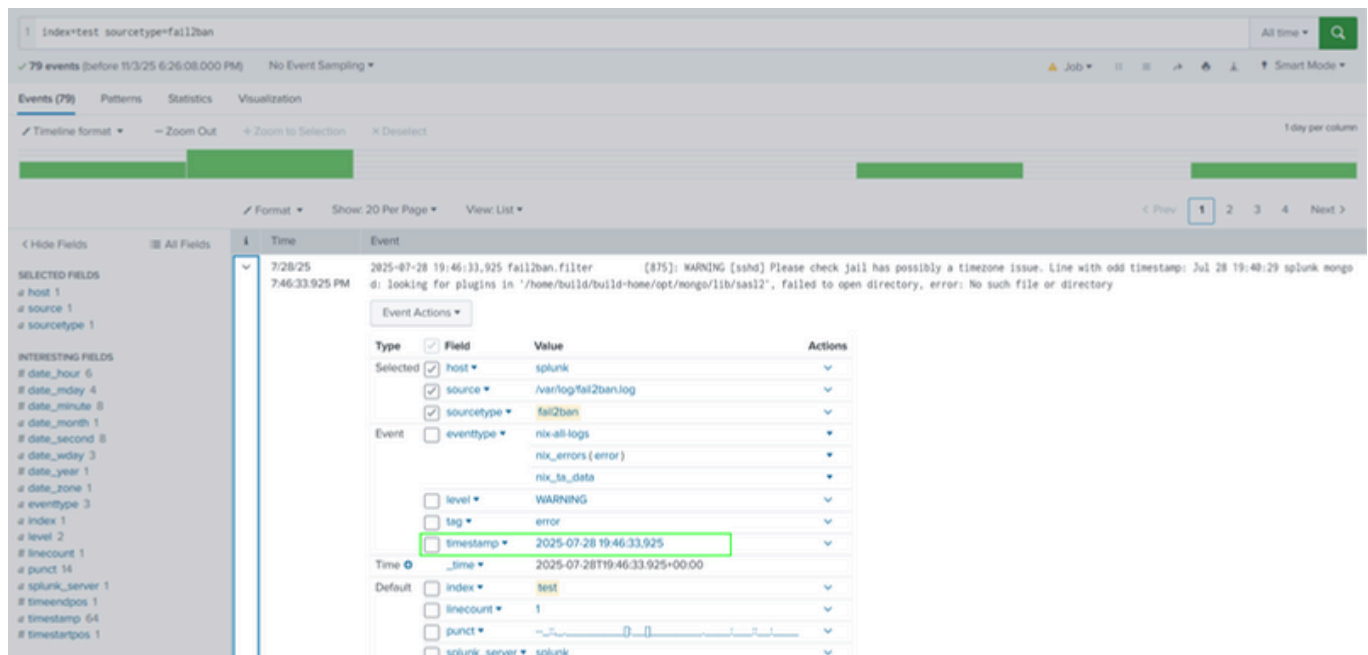
3. Restart Splunk to Apply Changes

```
splunk@splunk:~/splunk/etc/apps/fail2ban/default$ sudo ../../../../bin/splunk restart
```

- Note: The relative path is used because you're deep inside the app directory.

4. Verify in Splunk Search

Search:



- You should now see fields like timestamp in the left sidebar under Selected Fields, confirming that your extraction is working!

