

Forward-Looking Statements



This presentation may contain forward-looking statements regarding future events, plans or the expected financial performance of our company, including our expectations regarding our products, technology, strategy, customers, markets, acquisitions and investments. These statements reflect management's current expectations, estimates and assumptions based on the information currently available to us. These forward-looking statements are not guarantees of future performance and involve significant risks, uncertainties and other factors that may cause our actual results, performance or achievements to be materially different from results, performance or achievements expressed or implied by the forward-looking statements contained in this presentation.

For additional information about factors that could cause actual results to differ materially from those described in the forward-looking statements made in this presentation, please refer to our periodic reports and other filings with the SEC, including the risk factors identified in our most recent quarterly reports on Form 10-Q and annual reports on Form 10-K, copies of which may be obtained by visiting the Splunk Investor Relations website at www.investors.splunk.com or the SEC's website at www.sec.gov. The forward-looking statements made in this presentation are made as of the time and date of this presentation. If reviewed after the initial presentation, even if made available by us, on our website or otherwise, it may not contain current or accurate information. We disclaim any obligation to update or revise any forward-looking statement based on new information, future events or otherwise, except as required by applicable law.

In addition, any information about our roadmap outlines our general product direction and is subject to change at any time without notice. It is for informational purposes only and shall not be incorporated into any contract or other commitment. We undertake no obligation either to develop the features or functionalities described, in beta or in preview (used interchangeably), or to include any such feature or functionality in a future release.

Splunk, Splunk> and Turn Data Into Doing are trademarks and registered trademarks of Splunk Inc. in the United States and other countries. All other brand names, product names or trademarks belong to their respective owners. © 2022 Splunk Inc. All rights reserved.

Forget Add-on Builder! Build Splunk Platform Apps with Config Explorer!

DEV1160B

Brett Adams
Cyber Analytics Specialist | Deloitte



splunk> .conf22



Brett Adams

Cyber Analytics Specialist | Deloitte

Why not Splunk® Add-on Builder?

Is a great starting point for small projects and first time developers.

Has limited checkpointing capabilities.

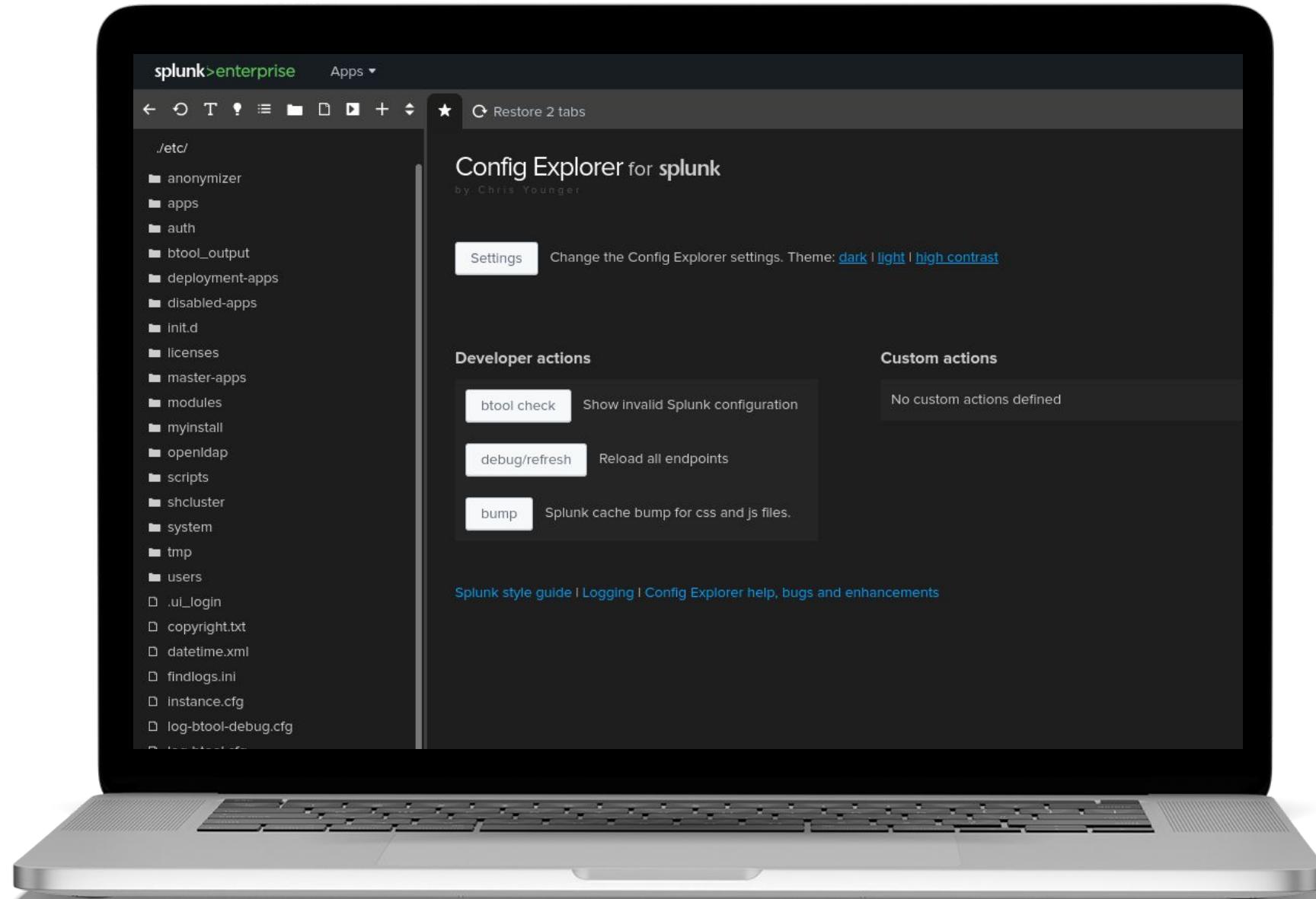
Creates large applications with multiple python dependencies.

By not using Add-on builder, you learn how Splunk apps actually work.

Config Explorer

Built by Chris Younger

Provides an editor interface for viewing and editing Splunk files.





Chris Younger

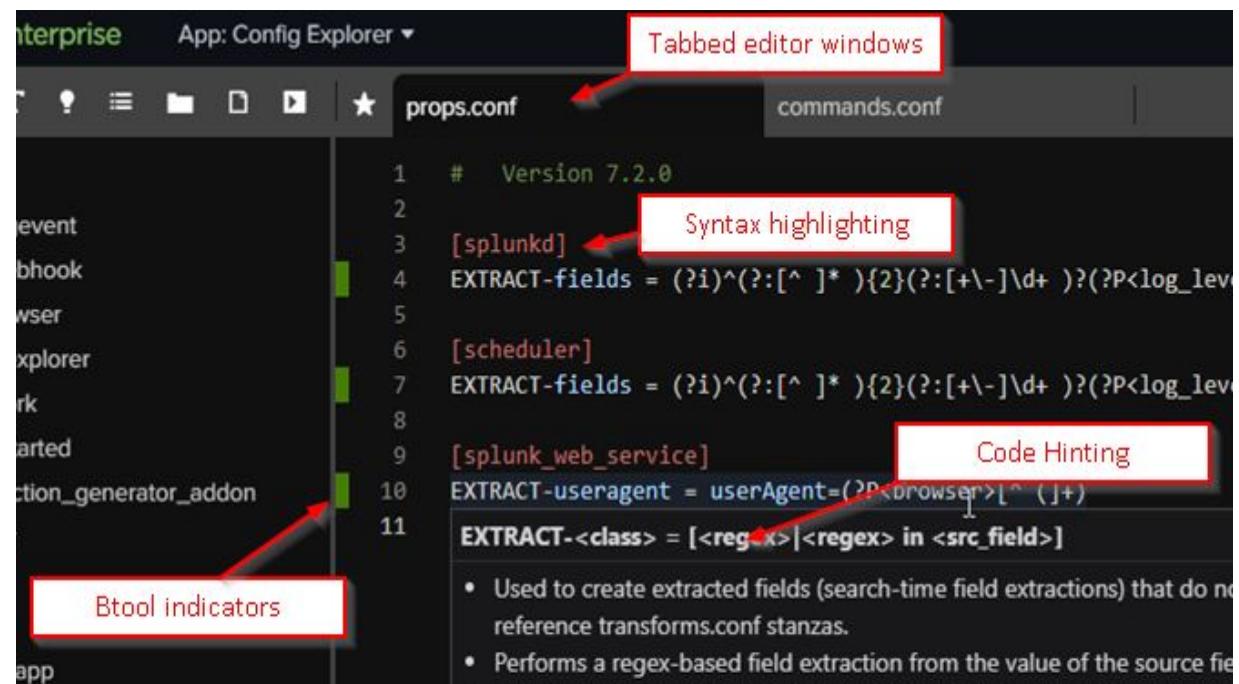
The Man, The Myth,
The Splunkbase Legend

[https://splunkbase.splunk.com/apps/#/product/all/
author/chrisyoungerjds](https://splunkbase.splunk.com/apps/#/product/all/author/chrisyoungerjds)

What is Config Explorer

<https://splunkbase.splunk.com/app/4353/>

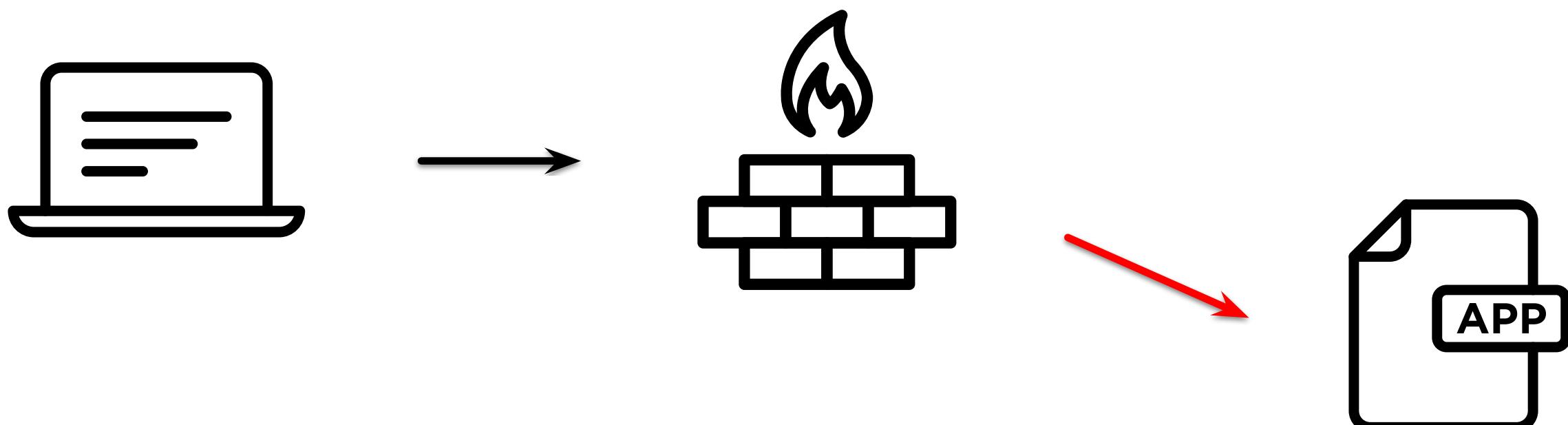
- Embeds the excellent Microsoft Monaco editor (VS Code)
- Code completion and tooltips
- Code gutter highlights using btool
- Displays the local spec files
- Diff files
- Works on Linux and Windows
- **Provides useful developer hooks (bump, debug refresh etc.) that can be customised and extended**
- **Optional version control of all changes by committing them to a git repository before and after changes**



The screenshot shows the Config Explorer application interface. At the top, there is a toolbar with icons for file operations and a dropdown menu labeled "App: Config Explorer". Below the toolbar, there are two tabs: "props.conf" (which is active) and "commands.conf". A red box highlights the tabs with the label "Tabbed editor windows". The main area is a code editor showing a configuration file. Syntax highlighting is applied to various tokens. A red box highlights the code "EXTRACT-fields = (?i)^(:[^]*){2}(:[+|-]\d+)?(?P<log_level>)" with the label "Syntax highlighting". Another red box highlights the section "[splunkd]" with the label "Btool indicators". On the right side of the editor, there is a "Code Hinting" panel. The panel contains a list of suggestions and descriptions, with a red box highlighting the first item: "Used to create extracted fields (search-time field extractions) that do not reference transforms.conf stanzas." Below this, another item is listed: "Performs a regex-based field extraction from the value of the source file".

The Magic

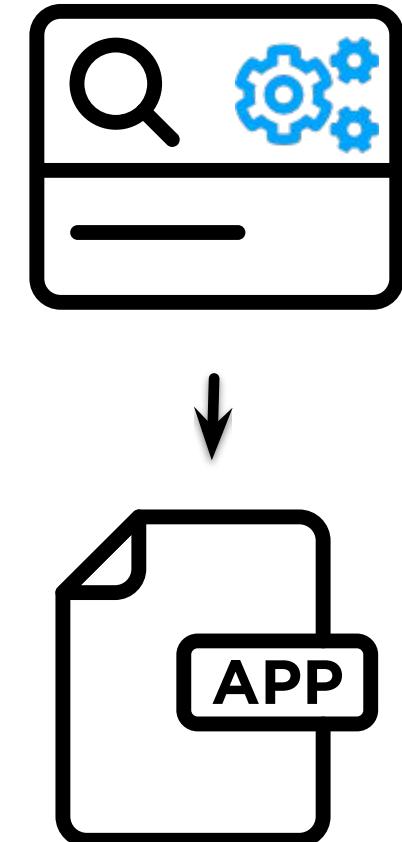
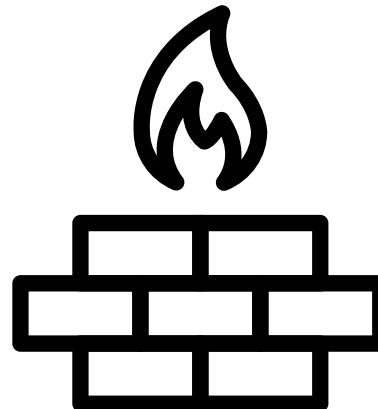
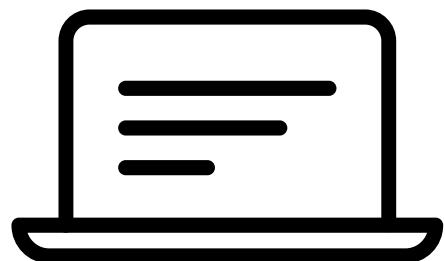
Remote Development over HTTPS



To learn more about developing with Visual Studio Code instead, check out [DEV1385](#) with Jason Conger.

The Magic

Remote Development over HTTPS



To learn more about developing with Visual Studio Code instead, check out [DEV1385](#) with Jason Conger.

Build in Enterprise

Run in Cloud & Enterprise





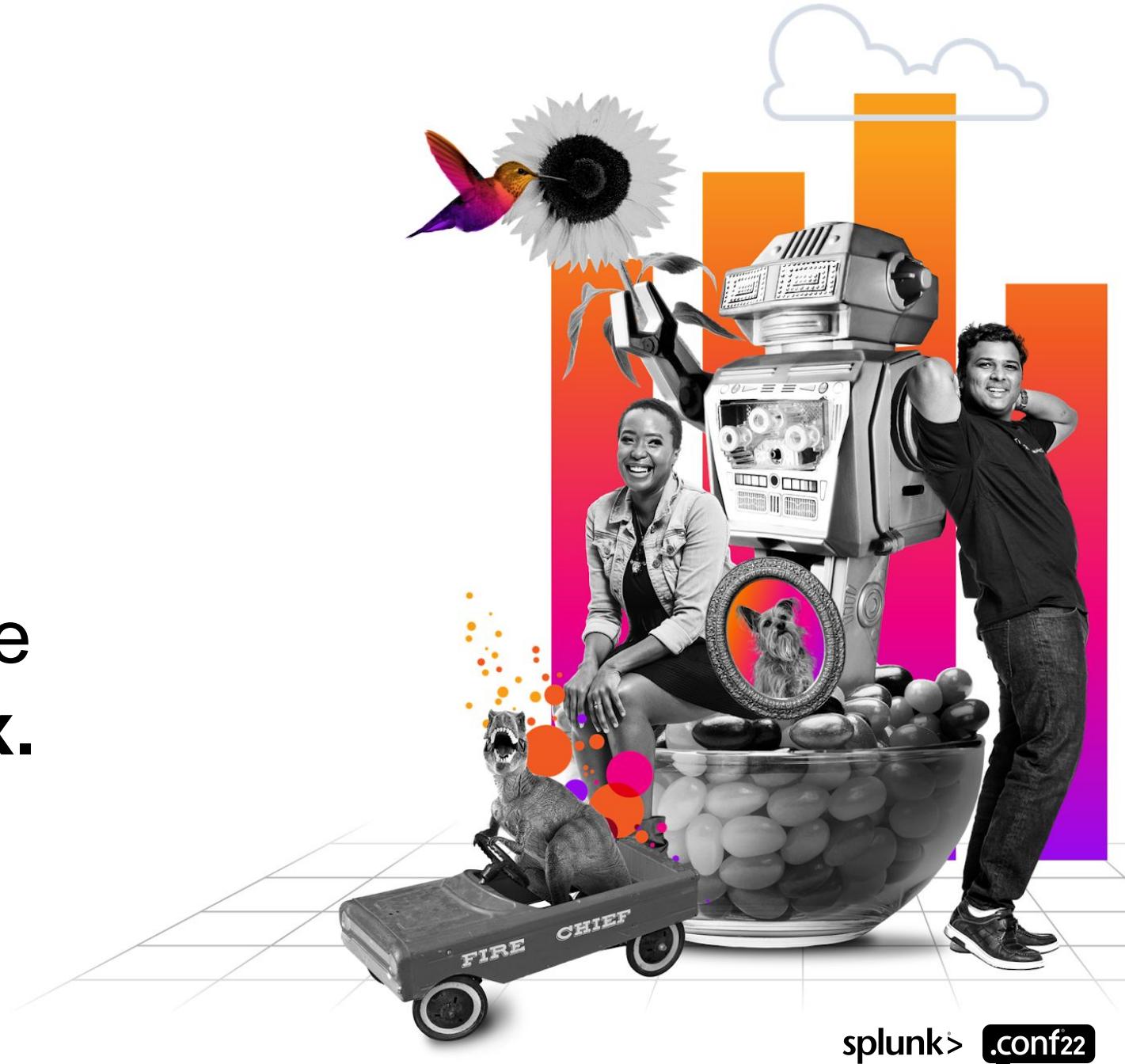
Development Resources

<https://dev.splunk.com/view/SP-CAAAER3>

https://github.com/splunk/splunk-sdk-python/tree/master/examples/github_commits

<https://github.com/LukeMurphey/splunk-modular-input>

I recommend
developing
Splunk Apps with
Splunk® Enterprise
running on Linux.



Let's Build an App!



Back button

Current directory

./etc/apps/TA_prismacloud_audit/

bin		08/04/2022, 09:58:32
default		08/04/2022, 10:01:10
lib		08/04/2022, 09:57:55
local		09/04/2022, 09:56:31
metadata		09/04/2022, 10:32:17
README		08/04/2022, 09:57:56
static		08/04/2022, 09:57:55
.gitignore	35 B	08/04/2022, 09:57:55
app.manifest	1.01 KB	08/04/2022, 10:46:27
README.md	230 B	08/04/2022, 10:04:01
splunkbaseid	0 B	08/04/2022, 10:04:05

Files and Folders

Config Explorer for splunk

by Chris Younger

Settings

Change the Config Explorer settings. Theme: [dark](#) | [light](#) | [high contrast](#)

Developer actions

btool check

Show invalid Splunk configuration

debug/refresh

Reload all endpoints

bump

Splunk cache bump for css and js files.

Custom actions

No custom actions defined

[Splunk style guide](#) | [Logging](#) | [Config Explorer help, bugs and enhancements](#)

splunk>enterprise Apps ▾

Administrator ▾ Messages ▾ Settings ▾ Activity ▾ Help ▾

Find

./etc/apps/TA_prismacloud_audit/

bin	08/04/2022, 09:58:32
default	08/04/2022, 10:01:10
lib	08/04/2022, 09:57:55
local	09/04/2022, 09:56:31
metadata	09/04/2022, 10:32:17
README	08/04/2022, 09:57:56
static	08/04/2022, 09:57:55
.gitignore	35 B 08/04/2022, 09:57:55
app.manifest	1.01 KB 08/04/2022, 10:46:27
README.md	230 B 08/04/2022, 10:04:01
splunkbaseid	0 B 08/04/2022, 10:04:05

Config Explorer for splunk
by Chris Younger

Settings Change the Config Explorer settings. Theme: [dark](#) | [light](#) | [high contrast](#)

Opens the Config Explorer Settings

Developer actions

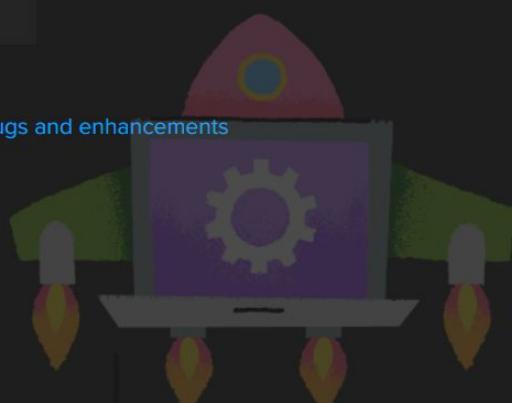
btool check Show invalid Splunk configuration

debug/refresh Reload all endpoints

bump Splunk cache bump for css and js files.

Custom actions

No custom actions defined



[Splunk style guide](#) | [Logging](#) | [Config Explorer help, bugs and enhancements](#)

splunk>enterprise Apps ▾

Administrator ▾ Messages ▾ Settings ▾ Activity ▾ Help ▾

Find

Settings

./etc/apps/TA_prismacloud_audit/

bin
default
lib
local
metadata
README
static
.gitignore
app.manifest
README.md
splunkbaseid

[global]

3 **write_access = true**

4 # This allows the user to creating new files/folders, deleting and renaming. This
5 # is obviously very dangerous and just like having filesystem access through
6 # the operating system, will make it very easy to destroy your Splunk
7 # environment if you dont know what you are doing.
8 # Defaults to false

10 **run_commands = true**

11 # Run commands in the shell, with the user that splunk runs as.
12 # Use with caution.
13 # Defaults to false

15 **hide_settings = false**

16 # Hide the "Settings" link from the home screen. Note that if write_access
17 # is true then settings can still be changed at
18 # etc/apps/config_explorer/local/config_explorer.conf. When the Settings link
19 # is displayed, it can be changed even when write_access is off. To
20 # prevent all editing, set hide_settings = true and write_access = false .
21 # Defaults to false.

23 **#max_file_size = 10**

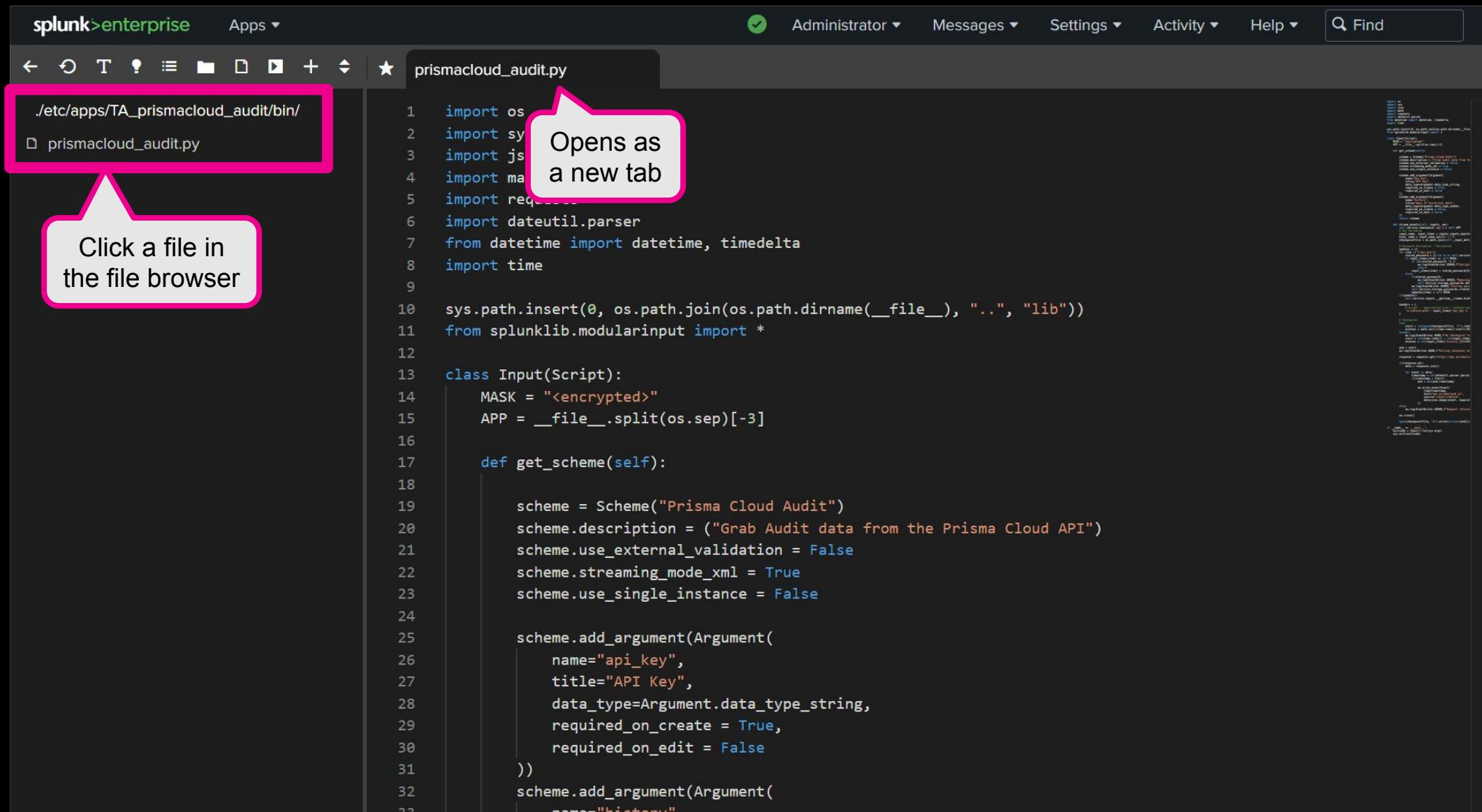
24 # The maximum file size in megabytes that can be opened.

26 **#cache_file_depth = 6**

27 # Cache the list of files and folders for the left pane to this many levels deep.
28 # This makes navigation much faster (especially on windows) but uses more memory,
29 # causes slightly slower startup, and will not follow symbolic links. Set to 0 to
30 # disable cache but allow caching of visited directories. Set -1 to disable all caching.
31 # Defaults to 5

33 **#conf validate on save = true**

Change both to 'true' then save with CTRL+S



Click a file in the file browser

Opens as a new tab

```
1  import os
2  import sys
3  import json
4  import math
5  import requests
6  import dateutil.parser
7  from datetime import datetime, timedelta
8  import time
9
10 sys.path.insert(0, os.path.join(os.path.dirname(__file__), "..", "lib"))
11 from splunklib.modularinput import *
12
13 class Input(Script):
14     MASK = "<encrypted>"
15     APP = __file__.split(os.sep)[-3]
16
17     def get_scheme(self):
18
19         scheme = Scheme("Prisma Cloud Audit")
20         scheme.description = ("Grab Audit data from the Prisma Cloud API")
21         scheme.use_external_validation = False
22         scheme.streaming_mode_xml = True
23         scheme.use_single_instance = False
24
25         scheme.add_argument(Argument(
26             name="api_key",
27             title="API Key",
28             data_type=Argument.data_type_string,
29             required_on_create = True,
30             required_on_edit = False
31         ))
32         scheme.add_argument(Argument(
33             name="history"
34         ))
```



Hooks

Being lazy for maximum productivity

./etc/apps/TA_prismacloud_audit/

- bin
- default
- lib
- local
- metadata
- README
- static
- .gitignore
- app.manifest
- README.md
- splunkbaseid

star icon

Prismacloud_audit.py Settings

```
[hook:python_scheme]
146 # when editing .py files in bin, have an option to run them as 'splunk cmd'
147 # This will only show if run_commands=true in the config file
148 match = .*/bin/.*\.py$ 
149 label = Run Scheme in Splunk env
150 action = run:splunk cmd python ${FILE} --scheme

[hook:python2_scheme]
154 # when editing .py files in bin, have an option to run them as 'splunk cmd'
155 # This will only show if run_commands=true in the config file
156 match = .*/bin/.*\.py$ 
157 label = Run Scheme with Python 2
158 action = run:splunk cmd python2 ${FILE} --scheme

160 # when editing .conf files in "deployment-apps", have
161 # this is also useful on search head deployers. See "deployment-apps"
162 # [hook:btool-deployment-apps-for-btool]
163 # match = /deployment-apps.*(?:local|default)/[^/]*\.
164 # label = [deployment-apps-for-btool] Run btool on ${FILE}
165 # action = btool:${BASEFILE}:/opt/splunk/etc/deployment-apps
166 # showWithSave = false

167 #####
168 # Custom home-tab actions
169 #####
170 #####
171 # Actions are buttons on the home tab that can be used
172 # the run_commands option must be enabled. Read the
173 # uses the same command options.
174 #####
175 # Examples below:
176 #####
177
```

Save file

Open documentation (.spec file)

Show out-of-the-box hooks

Change All Occurrences Ctrl+F2

Cut

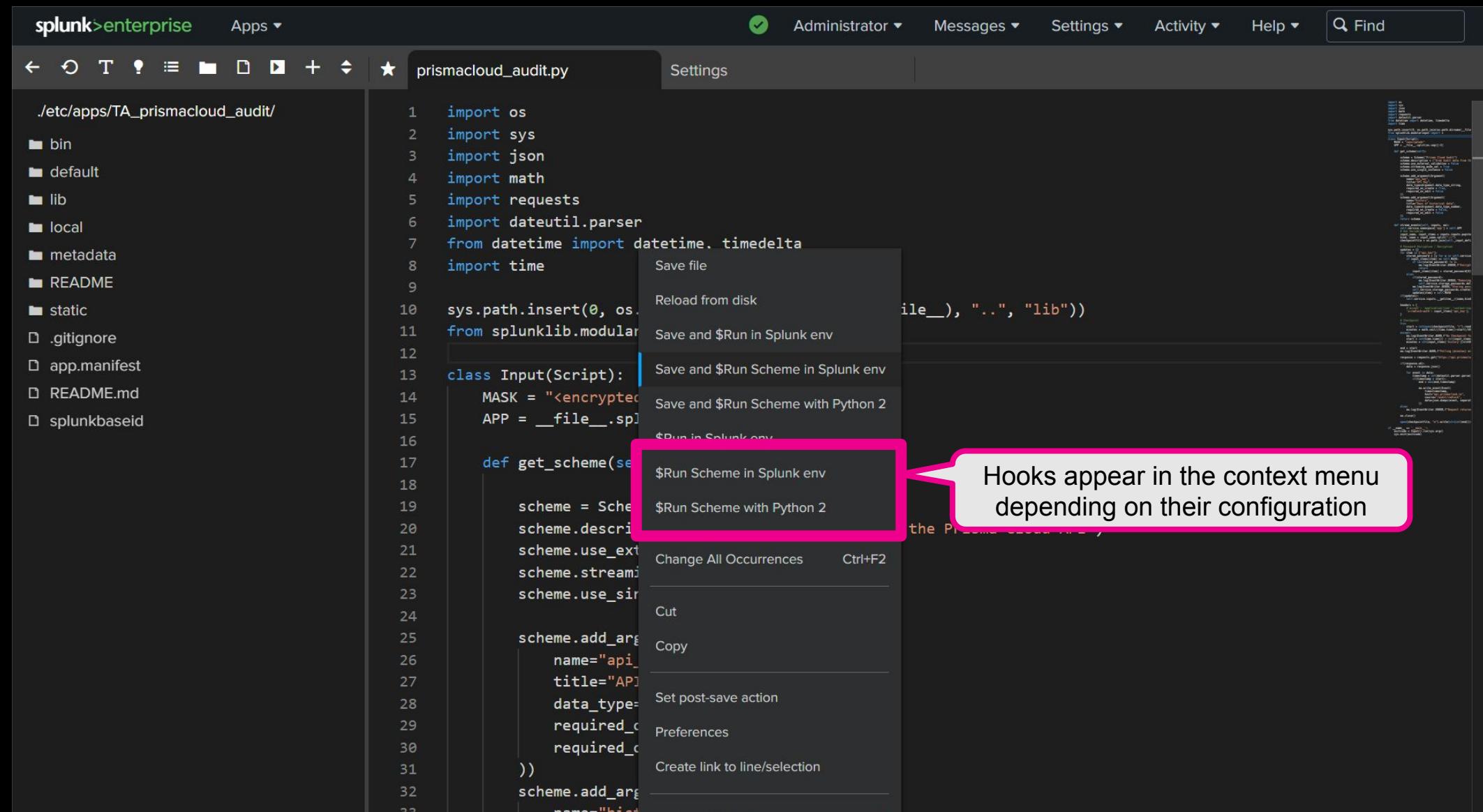
Copy

Set post-save action

Preferences

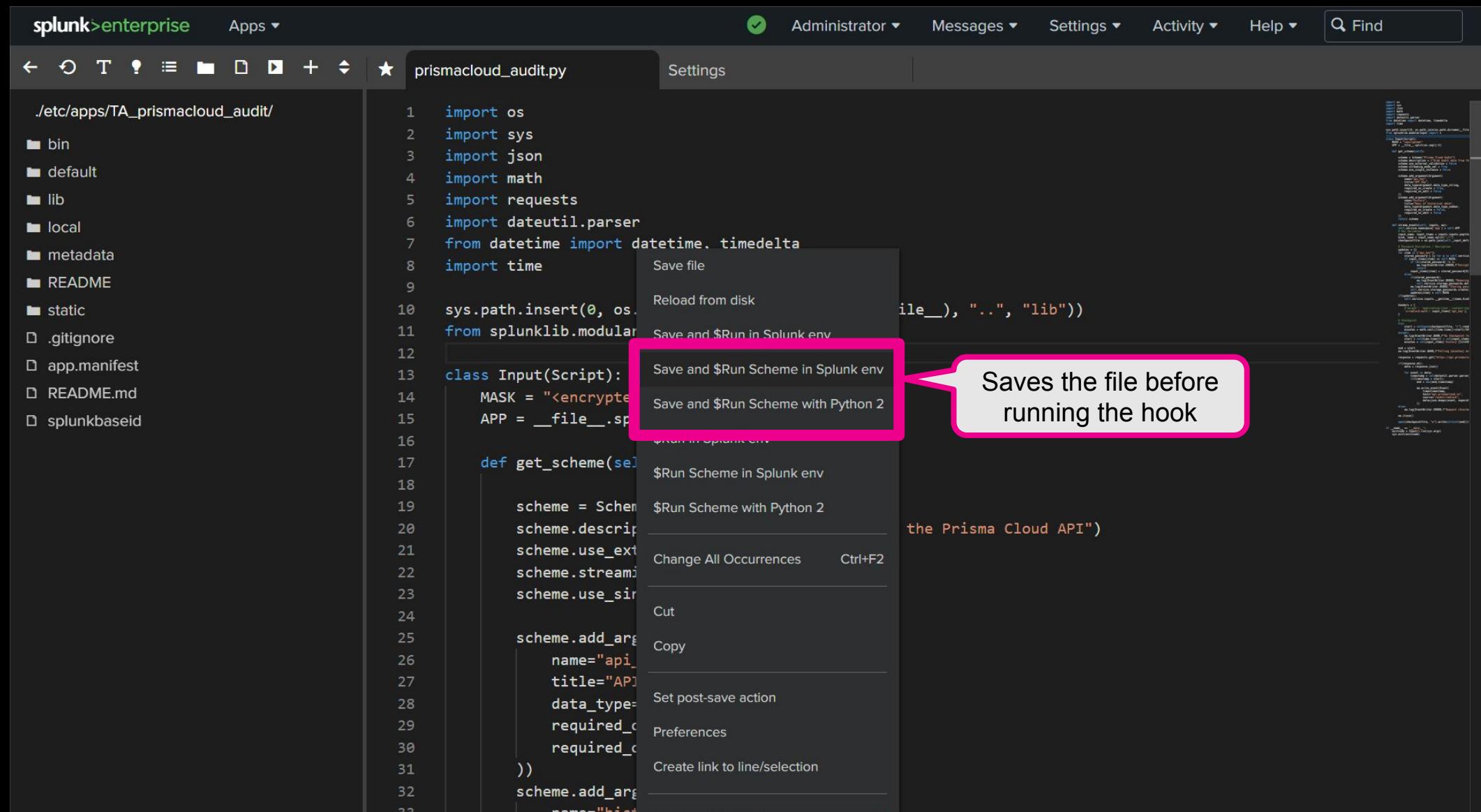
Create link to line/selection

Command Palette F1



The screenshot shows the Splunk Enterprise interface with the title bar "splunk>enterprise" and a user "Administrator". The main area displays a file named "prismacloud_audit.py" in a code editor. The code is a Python script for a TA (Trait Application) named "prismacloud_audit". A context menu is open over the code, showing several options. Two options are highlighted with a pink box: "\$Run Scheme in Splunk env" and "\$Run Scheme with Python 2". A pink callout bubble points from these highlighted options to a text box containing the following text: "Hooks appear in the context menu depending on their configuration".

```
1  import os
2  import sys
3  import json
4  import math
5  import requests
6  import dateutil.parser
7  from datetime import datetime, timedelta
8  import time
9
10 sys.path.insert(0, os.path.dirname(file__))
11 from splunklib.modular import Input
12
13 class Input(Script):
14     MASK = "<encrypted>"
```



./etc/apps/TA_prismacloud_audit/

bin
default
lib
local
metadata
README
static
.gitignore
app.manifest
README.md
splunkbaseid

prismacloud_audit.py

Administrator Messages Settings Activity Help Find

Save file
Reload from disk
Save and \$Run in Splunk env
Save and \$Run Scheme in Splunk env
Save and \$Run Scheme with Python 2
Save in Splunk env
\$Run Scheme in Splunk env
\$Run Scheme with Python 2
Change All Occurrences Ctrl+F2
Cut
Copy
Set post-save action
Preferences
Create link to line/selection

the Prisma Cloud API")

1 import os
2 import sys
3 import json
4 import math
5 import requests
6 import dateutil.parser
7 from datetime import datetime, timedelta
8 import time
9
10 sys.path.insert(0, os.
11 from splunklib.modular.
12
13 class Input(Script):
14 MASK = "<encrypted>"
15 APP = __file__.sp
16
17 def get_scheme(se
18
19 scheme = Sch
20 scheme.descri
21 scheme.use_ext
22 scheme.streami
23 scheme.use_sir
24
25 scheme.add_arg
26 name="api_

27 title="APJ

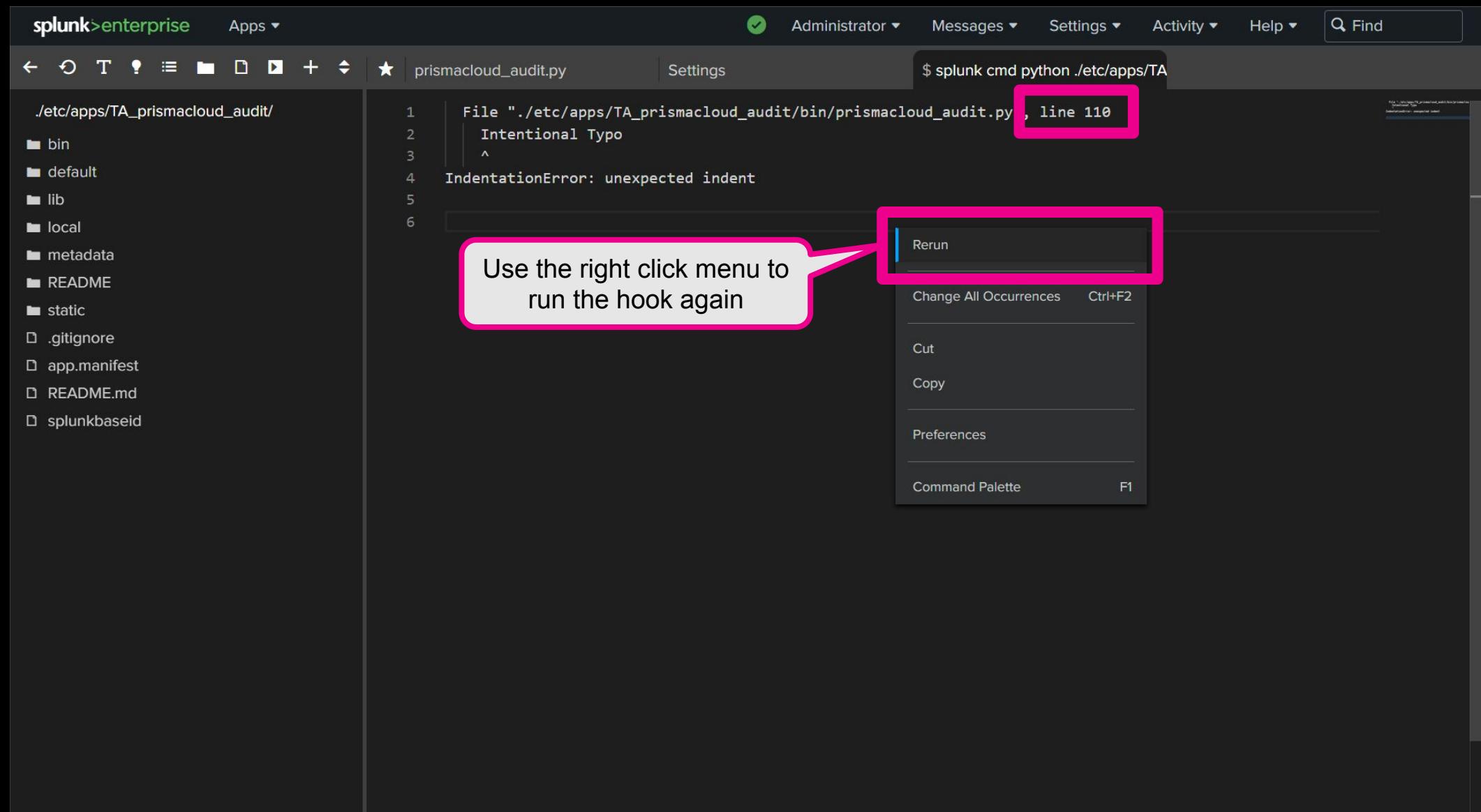
28 data_type="

29 required_c

30 required_c

31))

32 scheme.add_arg
33 name="hist



The screenshot shows the Splunk Enterprise interface with a code editor open. The file path is `./etc/apps/TA_prismacloud_audit/bin/prismacloud_audit.py`. The code editor displays the following text:

```
1  File "./etc/apps/TA_prismacloud_audit/bin/prismacloud_audit.py", line 110
2      Intentional Typo
3      ^
4  IndentationError: unexpected indent
5
6
```

A context menu is open over the line of code at line 110, specifically over the text "Intentional Typo". The menu items are:

- Rerun
- Change All Occurrences Ctrl+F2
- Cut
- Copy
- Preferences
- Command Palette F1

A callout bubble with the text "Use the right click menu to run the hook again" points to the "Rerun" option in the menu.

splunk>enterprise Apps ▾

Administrator ▾ Messages ▾ Settings ▾ Activity ▾ Help ▾ Find

prismacloud_audit.py Settings \$ splunk cmd python ./etc/apps/TA

```
./etc/apps/TA_prismacloud_audit/
  bin
  default
  lib
  local
  metadata
  README
  static
  .gitignore
  app.manifest
  README.md
  splunkbaseid
```

1 <schema><title>Prisma Cloud Audit</title><description>Grab Audit data from the Prisma Cloud API</description>

2

This is the expected
XML output

splunk>enterprise Apps ▾

Administrator ▾ Messages ▾ Settings ▾ Activity ▾ Help ▾

Find

prismacloud_audit.py

./etc/apps/TA_prismacloud_audit/bin/

prismacloud_audit.py

```
1  import os
2  import sys
3  import json
4  import math
5  import requests
6  import dateutil.parser
7  from datetime import datetime, timedelta
8  import time
9
10 sys.path.insert(0, os.path.join(os.path.dirname(__file__), 'lib'))
11 from splunklib.modularinput import *
12
13 class Input(Script):
14     MASK = "<encrypted>"
15     APP = __file__.split(os.sep)[-3]
16
17     def get_scheme(self):
18
19         scheme = Scheme("Prisma Cloud Audit")
20         scheme.description = ("Grab Audit data from Prisma Cloud")
21         scheme.use_external_validation = False
22         scheme.streaming_mode_xml = True
23         scheme.use_single_instance = False
24
25         string,
26         required_on_create = True,
27         required_on_edit = False
28     )
29     scheme.add_argument(Argument(
30         name="history"
31     ))
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
```

Save file

Reload from disk

Save and \$Run in Splunk env

Save and \$Run Scheme in Splunk env

Save and \$Run Scheme with Python 2

\$Run in Splunk env

\$Run Scheme in Splunk env

\$Run Scheme with Python 2

Change All Occurrences Ctrl+F2

Cut

Copy

Set post-save action

Create link to line/selection

Command Palette F1

Sets a post-save action for the open file

Click a suggestion to auto populate

Add success and failure strings

Set post-save action

Enter a command to automatically run after successful save of this file only. This action will be saved in your browser local storage (it will not affect other users or other browsers you use, but it will be remembered after browser refresh). Run commands will be executed from the SPLUNK_HOME directory.

File: `./etc/apps/TA_prismacloud_audit/bin/prismacloud_audit.py`

Suggest: `run:splunk cmd python2 ./etc/apps/TA_prismacloud_audit/bin/prismacloud_audit.py --scheme`

Suggest: `run:splunk cmd python ./etc/apps/TA_prismacloud_audit/bin/prismacloud_audit.py --scheme`

Run command `splunk cmd python ./etc/apps/TA_prismacloud_audit/bin/prismacloud_auc`

Run in background tab: Yes

The following options will check the returned content from a post-save action for a specific string and will show a success or failure icon in the file list. Only one string is required for the icon to be displayed.

Content match for success: `<schema>`

Content match for failure: `Error`

Save Cancel

```
28     data_type=Argument.data_type_string,
29     required_on_create = True,
30     required_on_edit = False
31   ))
32   scheme.add_argument(Argument(
33     name="history"
34   ))
```

Administrator Messages Settings Activity Help Find

prismacloud_audit.py \$ splunk cmd python ./etc/apps/TA_prismacloud_audit/bin/prismacloud_audit.py

Shows post save action was successful

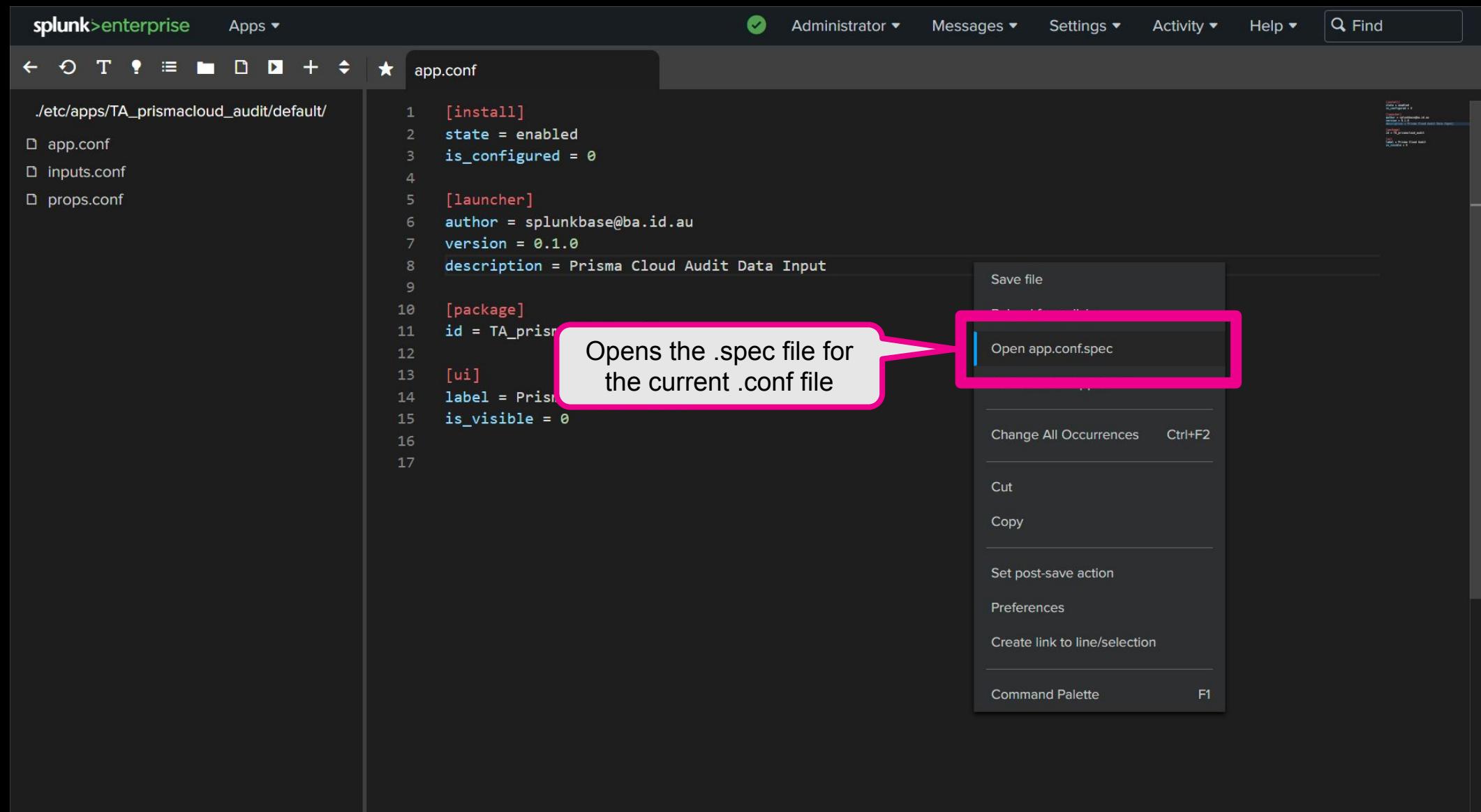
```
1  import os
2  import sys
3  import json
4  import math
5  import requests
6  import dateutil.parser
7  from datetime import datetime, timedelta
8  import time
9
10 sys.path.insert(0, os.path.join(os.path.dirname(__file__), "..", "lib"))
11 from splunklib.modularinput import *
12
13 class Input(Script):
14     MASK = "<encrypted>"
15     APP = __file__.split(os.sep)[-3]
16
17     def get_scheme(self):
18
19         scheme = Scheme("Prisma Cloud Audit")
20         scheme.description = ("Grab Audit data from the Prisma Cloud API")
21         scheme.use_external_validation = False
22         scheme.streaming_mode_xml = True
23         scheme.use_single_instance = False
24
25         scheme.add_argument(Argument(
26             name="api_key",
27             title="API Key",
28             data_type=Argument.data_type_string,
29             required_on_create = True,
30             required_on_edit = False
31         ))
32         scheme.add_argument(Argument(
33             name="history"
34         ))
```

 Saved



Spec Files

Better than typing inputs.conf.spec into Google



./etc/apps/TA_prismacloud_audit/default/

- app.conf
- inputs.conf
- props.conf

```
1  [install]
2  state = enabled
3  is_configured = 0
4
5  [launcher]
6  author = splunkbase@ba.id.au
7  version = 0.1.0
8  description = Prisma Cloud Audit Data Input
9
10 [package]
11 id = TA_prismacloud_audit
12
13 [ui]
14 label = Prisma Cloud Audit Data Input
15 is_visible = 0
16
17
```

Administrator Messages Settings Activity Help Find

Save file

Open app.conf.spec

Change All Occurrences Ctrl+F2

Cut

Copy

Set post-save action

Preferences

Create link to line/selection

Command Palette F1

Opens the .spec file for the current .conf file

splunk>enterprise Apps ▾

Administrator ▾ Messages ▾ Settings ▾ Activity ▾ Help ▾

Find

spec: app

./etc/apps/TA_prismacloud_audit/default/

app.conf inputs.conf props.conf

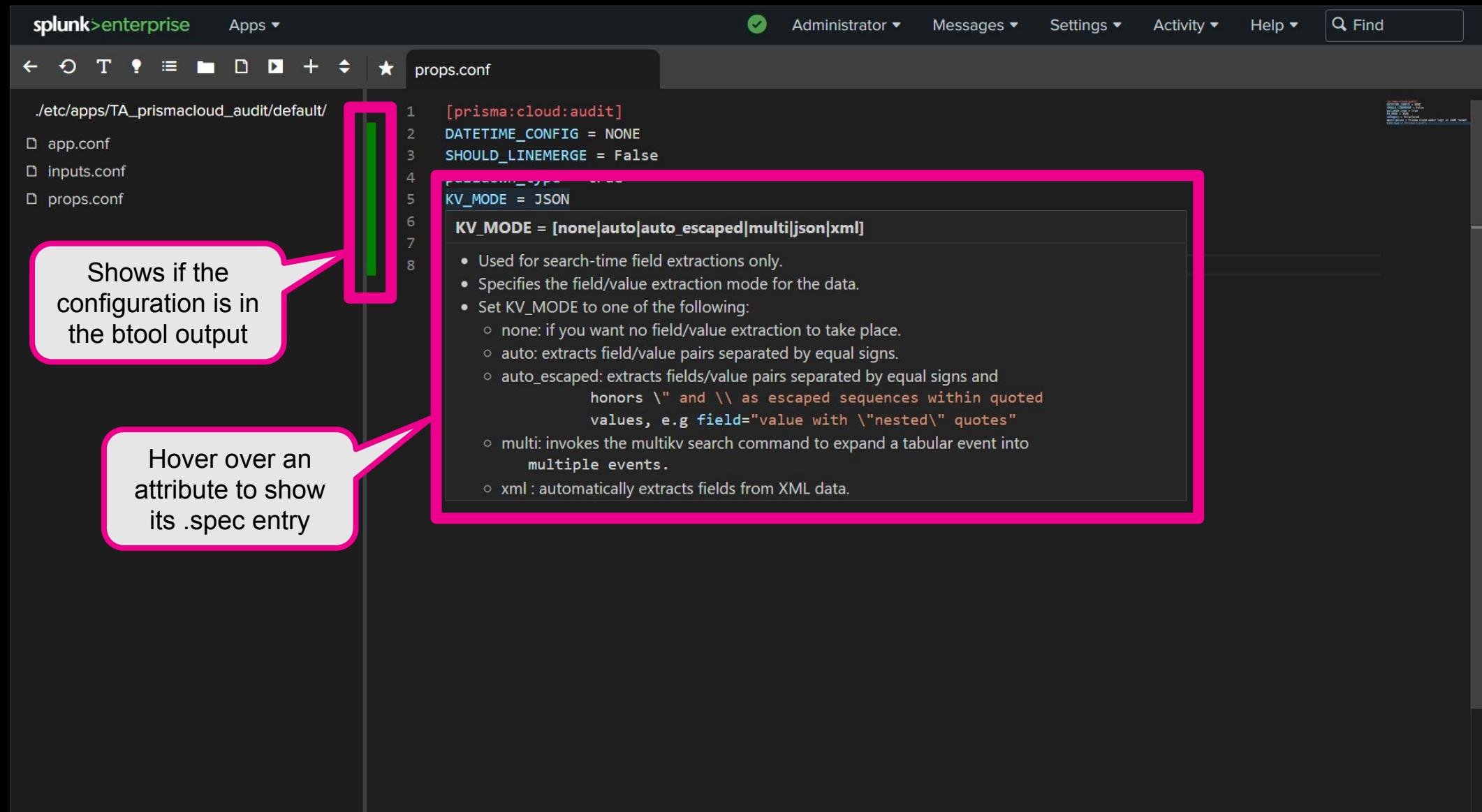
```
90
91 [package]
92 * This stanza defines upgrade-related metadata that streamlines app upgrade
93 to future versions of Splunk Enterprise.
94
95 id = <string>
96 * Omit this setting for apps that are for internal use only and not intended
97 for upload to Splunkbase.
98 * id is required for all new apps that you upload to Splunkbase. Future
99 Splunk Enterprise will use appid to correlate locally-installed apps
100 same app on Splunkbase (e.g. to notify users about app updates).
101 * id must be the same as the folder name in which your app lives in
102 $SPLUNK_HOME/etc/apps.
103 * id must adhere to these cross-platform folder name restrictions:
104 * must contain only letters, numbers, "." (dot), and "_" (underscore)
105 characters.
106 * must not end with a dot character.
107 * must not be any of the following names: CON, PRN, AUX, NUL,
108 COM1, COM2, COM3, COM4, COM5, COM6, COM7, COM8, COM9,
109 LPT1, LPT2, LPT3, LPT4, LPT5, LPT6, LPT7, LPT8, LPT9
110
111 check_for_updates = <boolean>
112 * Determines whether Splunk Enterprise checks for updates for this
113 app.
114 * Default: true
115
116 show_upgrade_notification = <boolean>
117 * Determines whether Splunk Enterprise shows an upgrade notification in Splunk
118 Web for this app.
119 * Default: false
120
121 [install]
122 * This stanza defines install settings for this app.
```

check_for_updates

Aa AbI * 1 of 1

Open Find with CTRL+F

Matches are highlighted



Shows if the configuration is in the btool output

Hover over an attribute to show its .spec entry

```
1 [prisma:cloud:audit]
2 DATETIME_CONFIG = NONE
3 SHOULD_LINEMERGE = False
4
5 KV_MODE = JSON
6 KV_MODE = [none|auto|auto_escaped|multi|json|xml]
7
8
• Used for search-time field extractions only.
• Specifies the field/value extraction mode for the data.
• Set KV_MODE to one of the following:
  ○ none: if you want no field/value extraction to take place.
  ○ auto: extracts field/value pairs separated by equal signs.
  ○ auto_escaped: extracts fields/value pairs separated by equal signs and
    honors \" and \\\" as escaped sequences within quoted
    values, e.g field="value with \"nested\" quotes"
  ○ multi: invokes the multikv search command to expand a tabular event into
    multiple events.
  ○ xml : automatically extracts fields from XML data.
```



Testing

Finding your faults, just like mom

splunk> .conf22

Testing

Outside Config Explorer

New Search

index=_internal component=ExecProcessor

136 of 224 events matched No Event Sampling ▾

Events (136) Patterns Statistics Visualization

Format Timeline ▾ – Zoom Out + Zoom to Selection X Deselect

1 second per column

1 2 3 4 5 6 7 Next >

Time	Event
09/04/2022 09:56:31.906	04-09-2022 09:56:31.906 +1000 ERROR ExecProcessor [5788 ExecProcessor] - message from "/opt/splunk/bin/python3.7 /opt/splunk/etc/apps/TA_prismacloud_audit/bin/prismacloud_audit.py" Request returned status 401 source = /opt/splunk/var/log/splunk/splunkd.log sourcetype = splunkd
09/04/2022 09:56:30.805	04-09-2022 09:56:30.805 +1000 WARN ExecProcessor [5788 ExecProcessor] - message from "/opt/splunk/bin/python3.7 /opt/splunk/etc/apps/TA_prismacloud_audit/bin/prismacloud_audit.py" No Checkpoint found, starting 90 days ago source = /opt/splunk/var/log/splunk/splunkd.log sourcetype = splunkd
09/04/2022 09:56:30.805	04-09-2022 09:56:30.805 +1000 WARN ExecProcessor [5788 ExecProcessor] - message from "/opt/splunk/bin/python3.7 /opt/splunk/etc/apps/TA_prismacloud_audit/bin/prismacloud_audit.py" Pulling 129600 minutes of data source = /opt/splunk/var/log/splunk/splunkd.log sourcetype = splunkd
09/04/2022 09:56:30.635	04-09-2022 09:56:30.635 +1000 INFO ExecProcessor [5788 ExecProcessor] - New scheduled exec process: /opt/splunk/bin/splunkd instrument-resource-usage

Prisma Cloud Audit

Data inputs > Prisma Cloud Audit

Showing 1-1 of 1 item

filter

25 per page ▾

name	API Key	Days of historical data	Source type	Index	Status	Actions
conf22	<encrypted>	90	prisma:cloud:audit	default	Enabled Disable	Clone Delete

Perfect Packaging

The shell script way

Remove compiled and cached python files

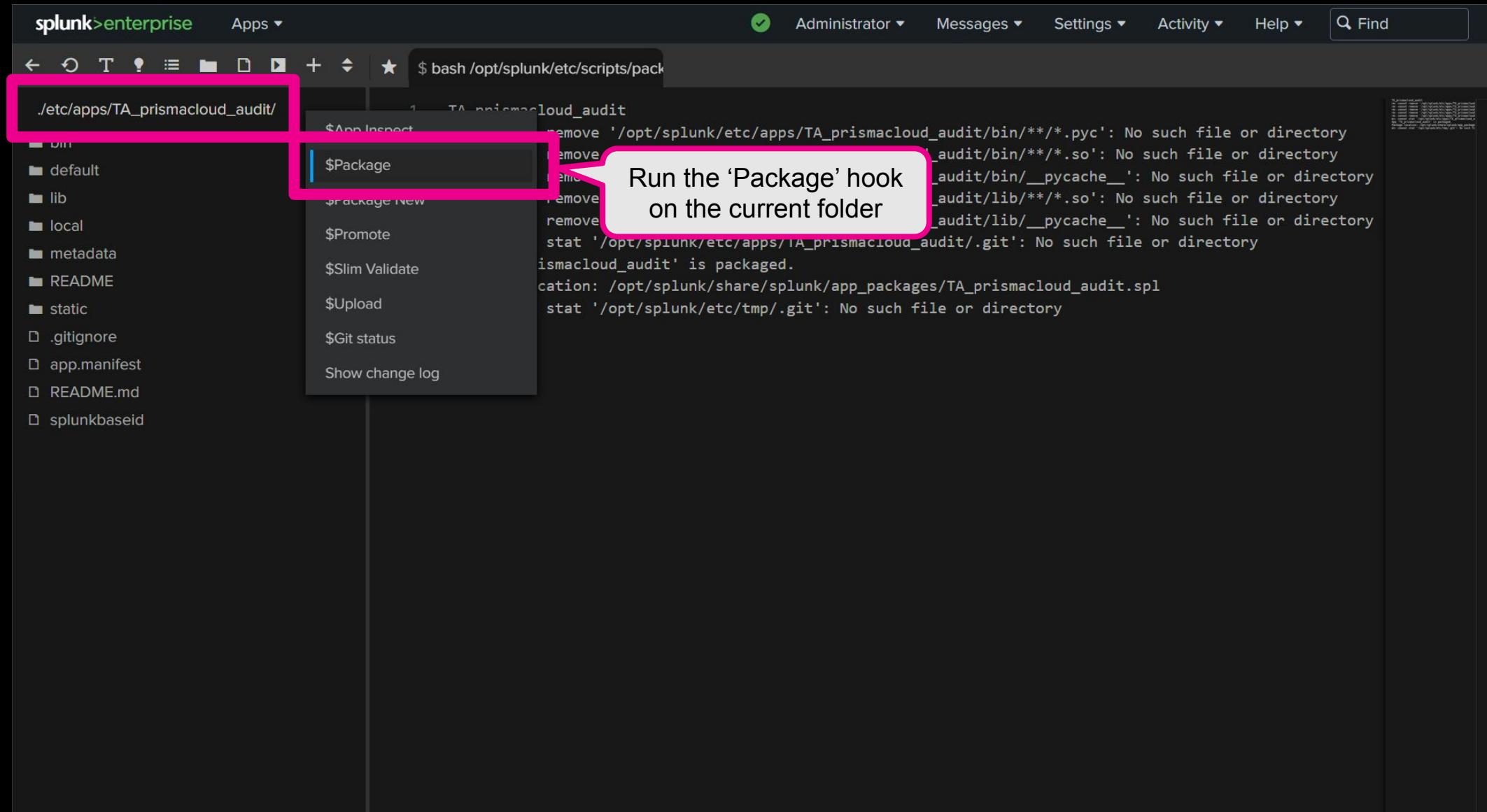
Remove metadata/local.meta

Remove any boilerplate files and Lookup Editor backups

Set all files to 644 (Owner Read & Write, Others Read)

Add execute permission to folders and files in bin

Compress to tar gzip archive with .spl extension



splunk>enterprise Apps ▾

Administrator ▾ Messages ▾ Settings ▾ Activity ▾ Help ▾ Q Find

High level outcome from AppInspect

Results from each specific tag

```
$ splunk cmd python ./etc/scripts/a
./etc/apps/TA_prismacloud_audit
  bin
  default
  lib
  local
  metadata
  README
  static
  .gitignore
  app.manifest
  README.md
  splunkbaseid
$App Inspect
$Package New
$Promote
$Slim Validate
$Upload
$Git status
Show change log
1 TA_prismacloud_audit
2 {'request_id': '1c209be3-da12-4
3 PROCESSING
4 SUCCESS
5 {'error': 0, 'failure': 0, 'skipped': 0, 'manual_check': 25, 'not_applicable': 130, 'warning': 3, 'succes
6
7 cloud
8 [{'result': 'warning', 'message': "28 Python files found. Update these Python scripts to be cross-compatibl
9 [{'result': 'manual_check', 'message': "Python script is not well formed, syntax error found in python sc
10 [{'result': 'manual_check', 'message': 'The following lines should be inspected during code review, Possibl
11 [{'result': 'manual_check', 'message': 'The following line will be inspected during code review. The `__b
12 [{'result': 'manual_check', 'message': 'The following lines should be inspected during code review, `gzip
13 [{'result': 'manual_check', 'message': 'The following line will be inspected during code review. The `os.
14 [{'result': 'manual_check', 'message': 'The following lines should be inspected during code review. `socket
15 [{'result': 'warning', 'message': 'Bias language is found in the app. ent.<<<WHITELIST>>> (lib/splunklib/ent
16 [{'result': 'manual_check', 'message': 'Environment variable being used in lib/splunklib/searchcommands/v
17 [{'result': 'manual_check', 'message': 'The executable will be inspected during code review: File: lib/spl
18 self-service
19 [{'result': 'warning', 'message': "Json Schema version of 2.0.0 is not yet compatible with any versions o
20
21 future
22
23
24
25
```

splunk>enterprise Apps ▾

Administrator ▾ Messages ▾ Settings ▾ Activity ▾ Help ▾

Config Explorer for splunk
by Chris Younger

Change the Config Explorer settings. Theme: [dark](#) | [light](#) | [high contrast](#)

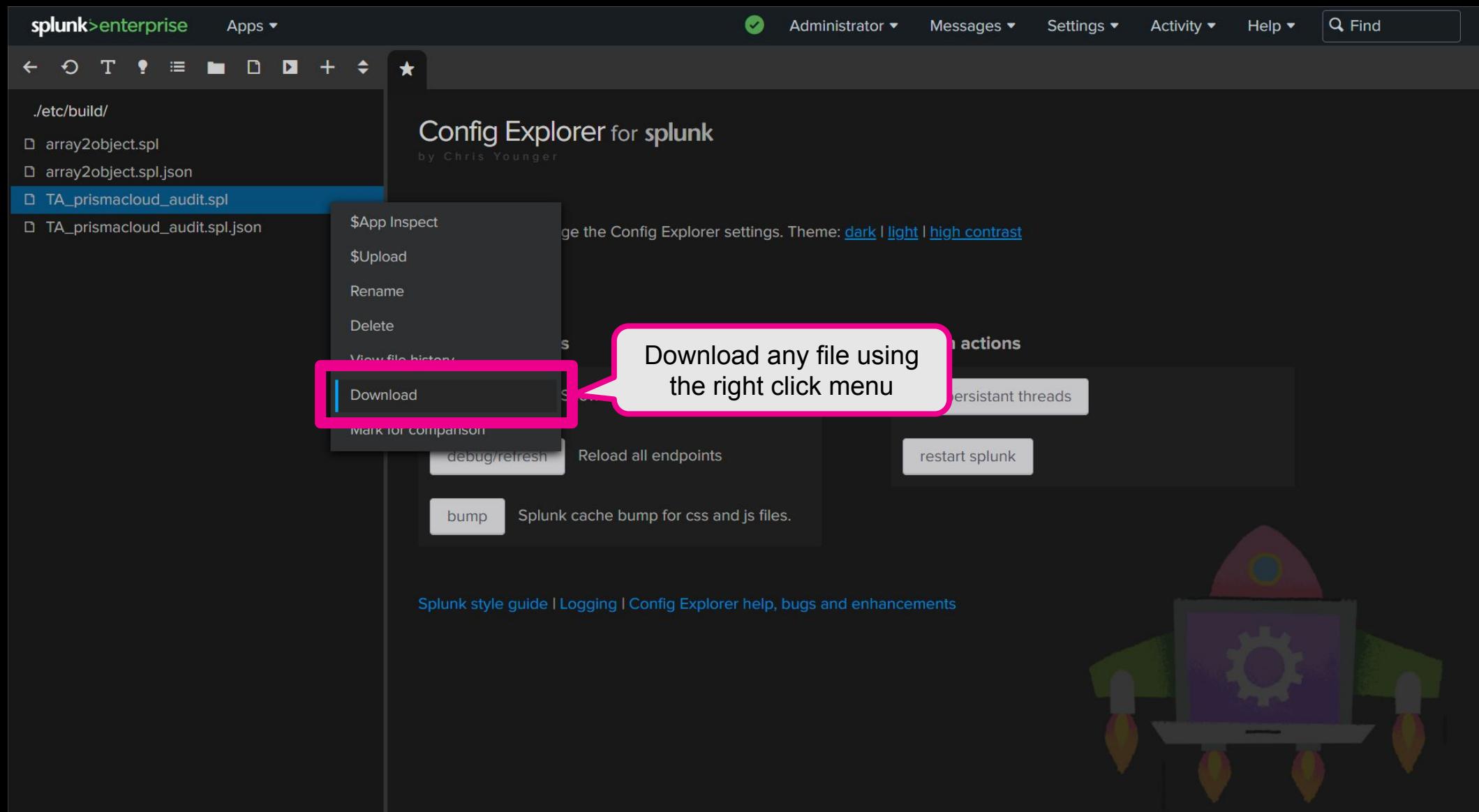
./etc/build/

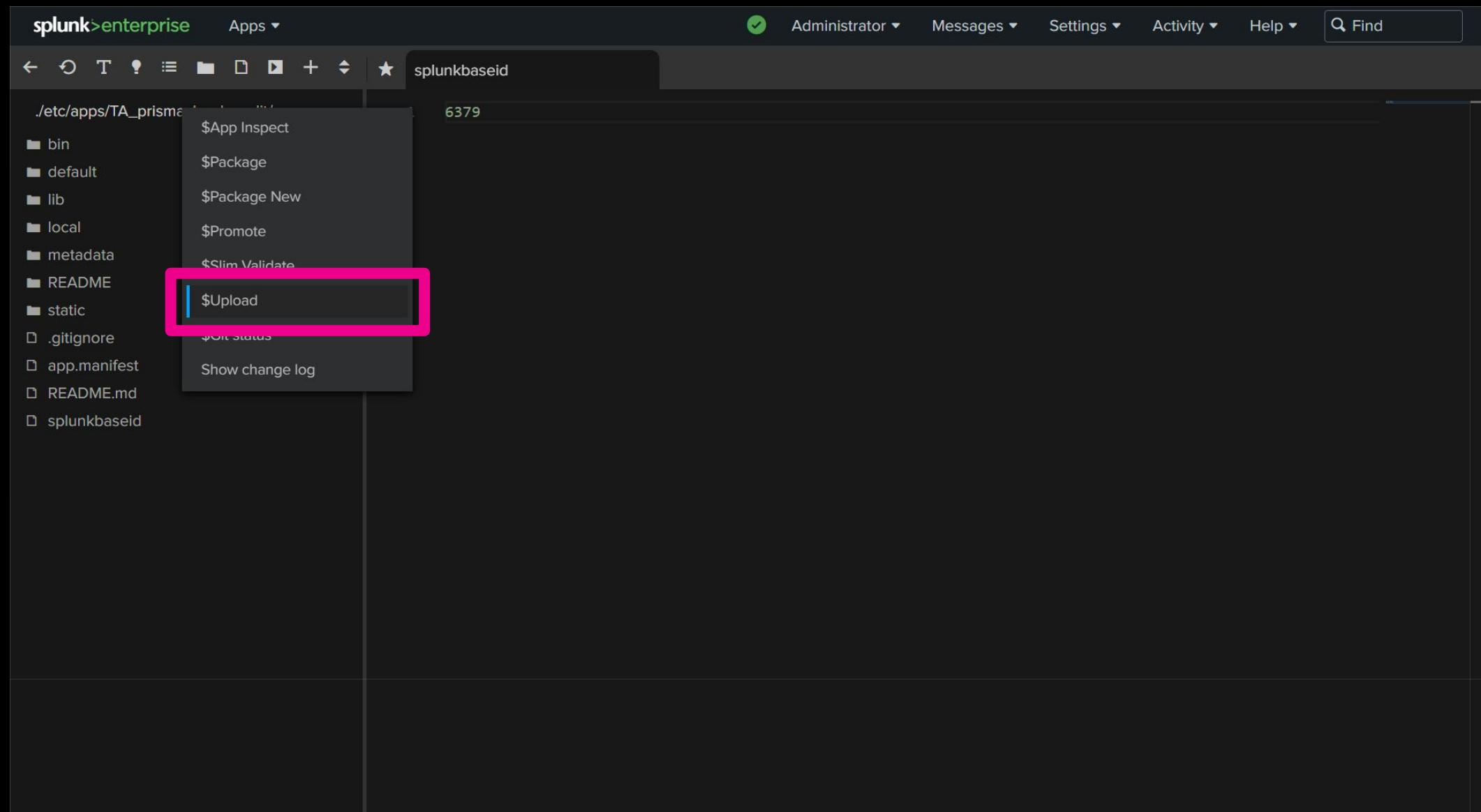
- array2object.spl
- array2object.json
- TA_prismacloud_audit.spl**
- TA_prismacloud_audit.spl.json

\$App Inspect
\$Upload
Rename
Delete
View file history
Download
Mark for comparison
debug/refresh Reload all endpoints restart splunk

bump Splunk cache bump for css and js files.

Download any file using the right click menu







Source Control

Keeping a permanent record of every mistake

splunk>enterprise Apps ▾

Administrator ▾ Messages ▾ Settings ▾ Activity ▾ Help ▾

Find

./etc/apps/TA_prismacloud_audit/

bin
default
lib
local
metadata
README
static
.gitignore
app.manifest
README.md
splunkbaseid

Settings

```
50
51 git_autocommit = true
52 # Track all file saves by automatically committing them
53 # Note you must first configure git to be used
54 # Autocommitting is a 'best effort' operation
55 # Defaults to false
56
57 #git_autocommit_show_output = auto
58 # When autocommit is enabled, when should we show the commit log
59 # true = Always show git messages
60 # false = Never show git output
61 # auto = Only show git messages when there is a non-zero status code
62 # Defaults to auto
63
64 #git_autocommit_dir =
65 # Force specific git repository location, relative to SPLUNK_HOME directory.
66 # Defaults to empty, meaning normal git rules will apply (search up from current directory)
67
68 #git_autocommit_work_tree =
69 # Force root location from where changes are tracked, relative to SPLUNK_HOME directory
70 # Set to "etc/" to track all changes beneath etc folder.
71 # Defaults to empty, meaning the normal git behavior will apply.
72
73 #detect_changed_files = true
74 #* Check if files that are open have changed on the filesystem and warn if so.
75 #* Defaults to true
76
77 ######
78 # Custom action hooks
79 ######
80
81 # Custom hooks create right-click actions for files that match a regular expression
82 # See etc/apps/config_explorer/default/config_explorer.conf for many examples
```

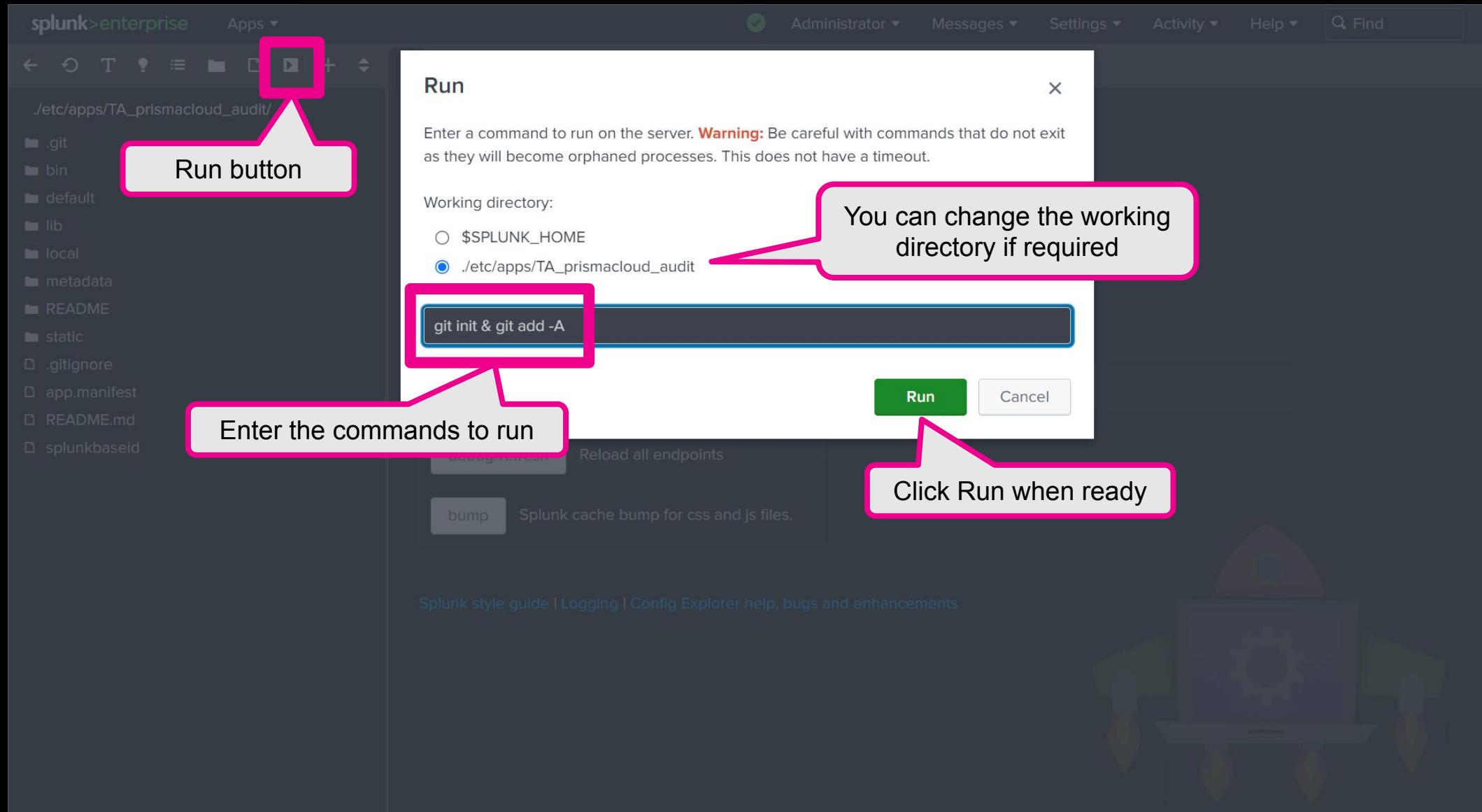
Change to true and don't forget to save

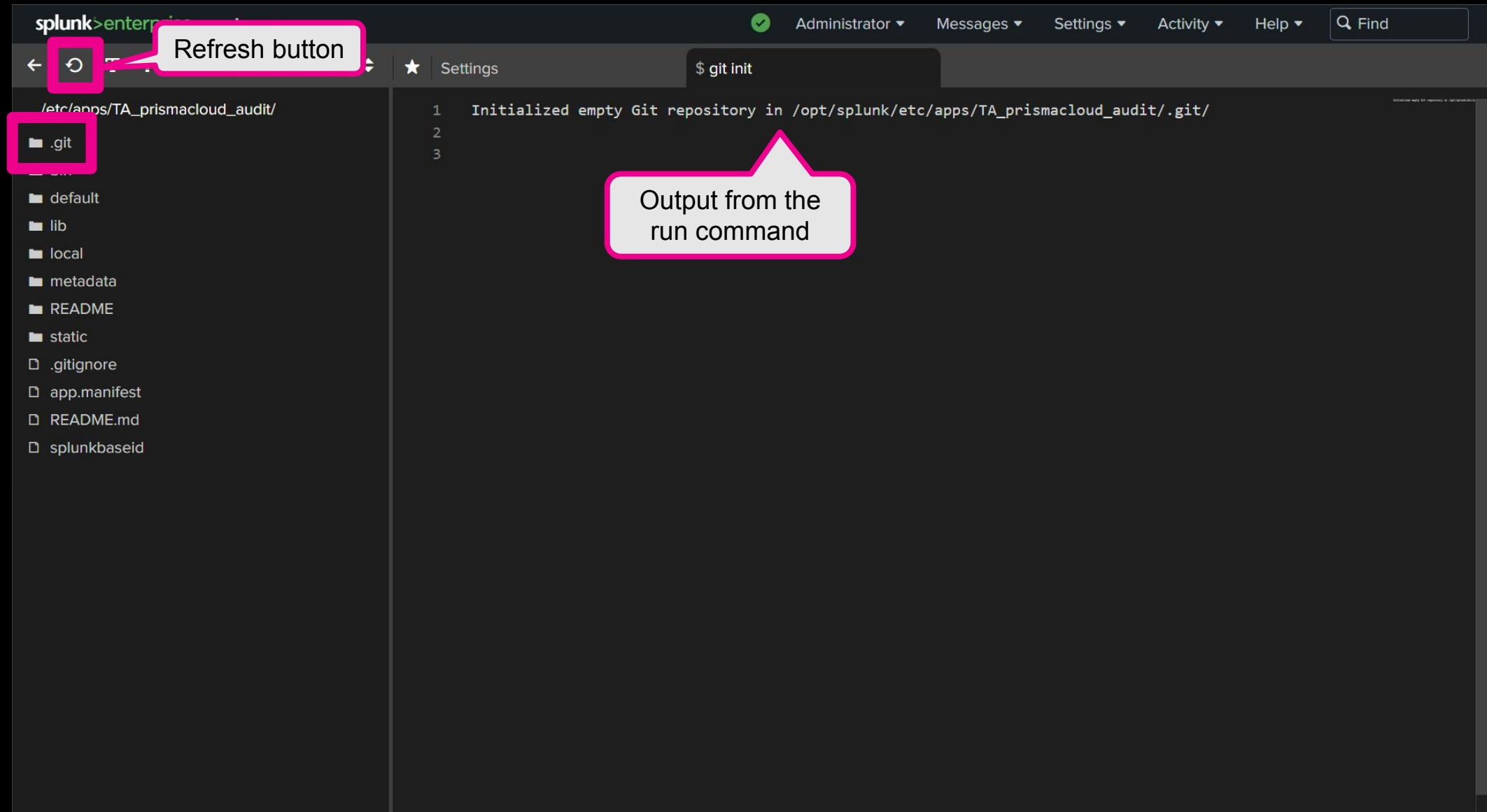
.gitignore

local/
local.meta
__pycache__
*.pyc

splunk> .conf22







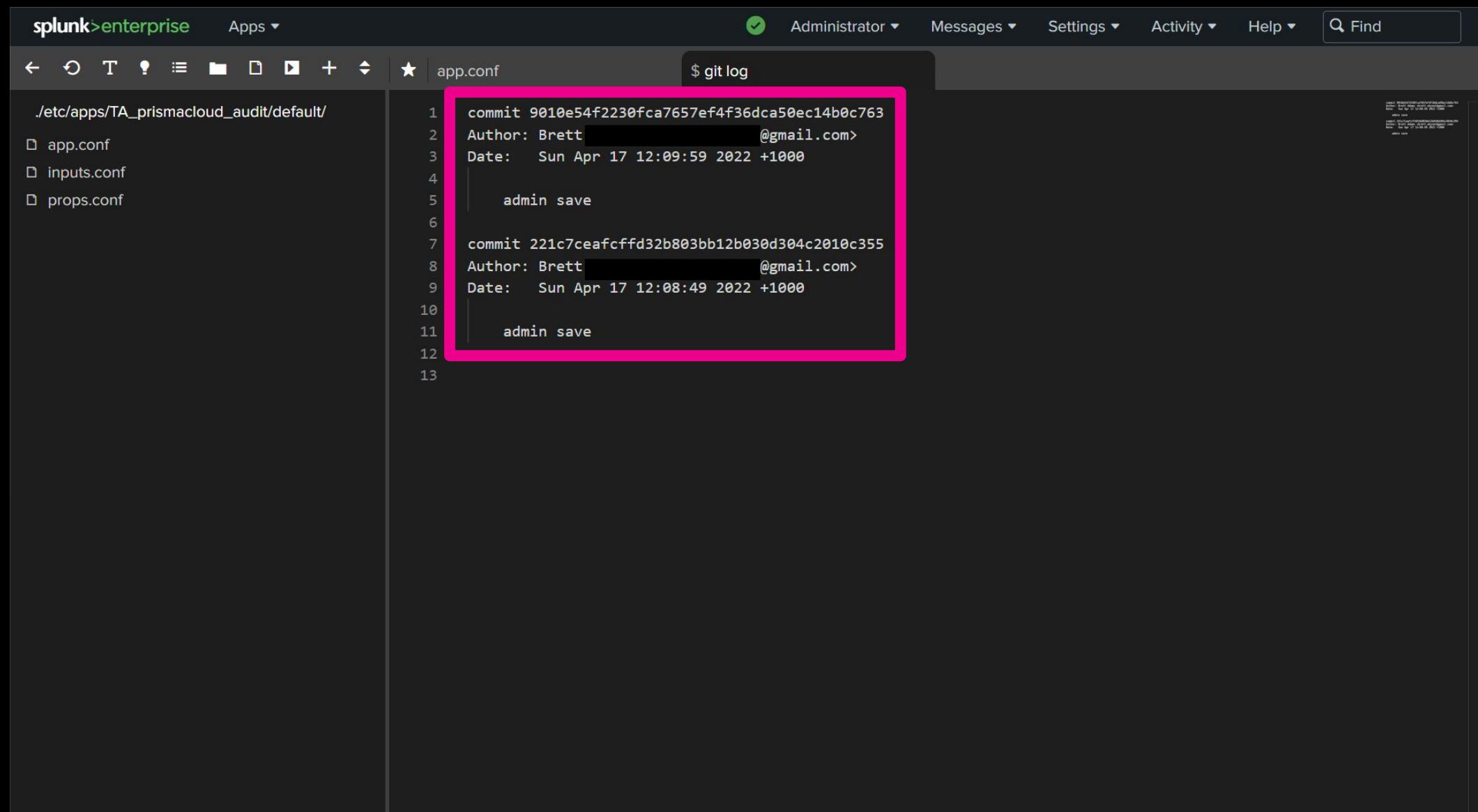
splunk>enterprise Apps ▾

Administrator ▾ Messages ▾ Settings ▾ Activity ▾ Help ▾

app.conf \$ git log

```
./etc/apps/TA_prismacloud_audit/default/
  □ app.conf
  □ inputs.conf
  □ props.conf

1 commit 9010e54f2230fc...a50ec14b0c763
2 Author: Brett [REDACTED] @gmail.com>
3 Date: Sun Apr 17 12:09:59 2022 +1000
4
5     admin save
6
7 commit 221c7ceafcffd32b803bb12b030d304c2010c355
8 Author: Brett [REDACTED] @gmail.com>
9 Date: Sun Apr 17 12:08:49 2022 +1000
10
11     admin save
12
13
```



Code Issues Pull requests Actions Projects Wiki Security Insights Settings

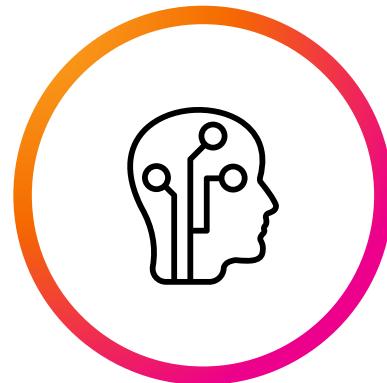
main

Commits on Jan 20, 2022

admin save	 Bre77 committed on 20 Jan	 a7c98ae 
admin save	 Bre77 committed on 20 Jan	 bbb0423 
admin save	 Bre77 committed on 20 Jan	 2c4a33c 
admin save	 Bre77 committed on 20 Jan	 15f44d1 
admin save	 Bre77 committed on 20 Jan	 e9d1672 
admin save	 Bre77 committed on 20 Jan	 51ea89b 
admin save	 Bre77 committed on 20 Jan	 bd8a217 
admin save	 Bre77 committed on 20 Jan	 29ae06b 
admin save	 Bre77 committed on 20 Jan	 9abac9c 
admin save		

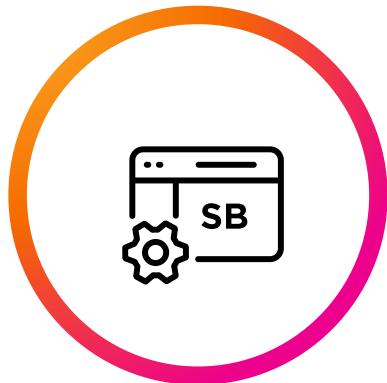
Recap

Create and Use Hooks



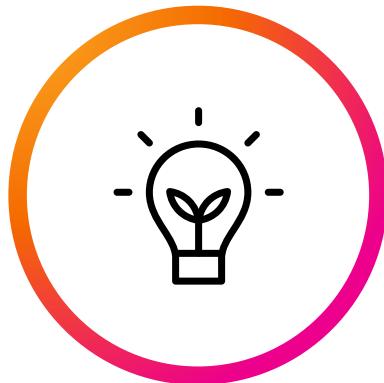
Speed up repetitive tasks

Reference the Spec & Btool



Avoid simple mistakes and write perfect conf

Test Fast and AppInspect



Using hooks, realtime search, and AppInspect

Easy Source Control



Publish your source (but not your passwords)



Git Hooked

On Config Explorer

<https://github.com/Bre77/DEV1160B>

Thank You

