# Table of Contents

**Introduction**
- About the Splunk Add-on Builder.................................................................1
- Learn more and get help..................................................................................2

**Get started**
- Install the Add-on Builder...............................................................................3
- Design your add-on..........................................................................................14

**Use the Add-on Builder**
- Use the Splunk Add-on Builder.......................................................................16
- Import and export an add-on project..............................................................16
- Create an add-on.............................................................................................18
- Configure data collection using a REST API call..........................................20
- Configure data collection using a shell command........................................25
- Configure data collection using your Python code.......................................27
- Create a setup page.........................................................................................29
- Add sample data............................................................................................31
- Manage source types......................................................................................32
- Extract fields..................................................................................................34
- Map to data model..........................................................................................36
- Create alert actions.........................................................................................38
- Validate and package.....................................................................................42

**Use the Add-ons**
- Use the add-on...............................................................................................45

**Advanced**
- Create a new data model................................................................................47
- Modify files directly.........................................................................................48
- Python helper functions..................................................................................49

**Release Notes**
- Release notes.................................................................................................52
- Third-party software credits...........................................................................52
Introduction

About the Splunk Add-on Builder

The Splunk Add-on Builder helps you build and validate add-ons for your Splunk platform deployment. The Splunk Add-on Builder is ideal for:

- Splunk admins who would like to onboard additional data into the Splunk platform.
- Developers who are looking for a tool to help them build and validate a Splunk add-on.

How an add-on helps you use the Splunk platform

An add-on is a reusable Splunk component that you build to perform a function in your Splunk deployment, such as getting a unique set of system data in and out of the Splunk software.

Add-ons can include any combination of

- custom configurations
- scripts
- data inputs
- custom reports or views
- themes that can change the look and feel of the Splunk platform.

Your add-on can be used in multiple apps, suites, or solutions.

Specialized add-ons help to collect, transform, and normalize data feeds from specific sources in your environment. Specialized add-ons are often referred to as Technology Add-ons, and can include some of the following functions:

- Knowledge management components that make the data easy to use. Examples include field extractions, transforms, and lookups,
- Knowledge mapping components that normalize the data to the **Common Information Model**. Examples include event types and tags,
- Configurations and/or tools that gather data from a source.

For more about apps and add-ons, see Develop Splunk Apps on the Splunk Developer Portal.

How the Splunk Add-on Builder can help you

The Splunk Add-on Builder guides you through the process of creating an add-on. The goals of the Splunk Add-on Builder are to:

- Guide you through all of the necessary steps of creating an add-on
- Build alert actions and adaptive response actions for Splunk Enterprise Security
- Reduce development and testing time
- Follow best practices and naming conventions
- Maintain CIM compliance
- Maintain quality of add-ons
- Validate and test the add-on, helping you to check for certification readiness and to identify any limitations such as compatibilities and dependencies
• Maintain a consistent look and facilitate branding
• Package the add-on and helps you get ready to submit it for certification

Learn more and get help

See the following resources to learn more about the Splunk Add-on Builder and get help if needed.

Support

For general Splunk platform support, see the Splunk Support Programs page.

If you have a support contract and have a specific question about the Splunk Add-on Builder, file a case using the Splunk Support Portal. See Support and Services.

Learn more about Splunk Enterprise

For more information about Splunk Enterprise, see:

• Splunk Enterprise documentation
• Splunk Answers
Get started

Install the Add-on Builder

To build your add-ons:

- Install the Add-on Builder onto a single-instance development environment, such as your laptop.
- Use your Add-on Builder instance to build your new add-ons.
- Deploy your new add-ons to your larger deployment.

Use your Add-on Builder instance in a single instance environment, then deploy the resulting add-on to a larger environment:

- Do not install the Splunk Add-on Builder app in a production Splunk environment with real users.
- Do not install the Splunk Add-on Builder app in a search head cluster or index cluster environment.
- Splunk Light is not supported.

Requirements

Requirements for the Splunk Add-on Builder:

- **Splunk Enterprise** For Add-on Builder version 3.0.0 and later, Splunk Enterprise 8.0.0 or later is required for Python 3 compatibility. Develop your add-ons with the Add-on Builder in a single-instance development environment such as a laptop.
  - Add-on Builder is not supported in a search head cluster or index cluster environment.
  - You can use the Add-on Builder Free with the Free license. Adding a Splunk Developer License lets you work with more data.
  - For more about Splunk Enterprise system requirements, see System Requirements in the Splunk Enterprise Installation Manual.
- **Mozilla Firefox, Google Chrome, or Microsoft Internet Explorer 11**
- **Admin privileges on Splunk Enterprise**. You must have Admin privileges to create an add-on project and to create and run modular inputs for data collection.
- **The Splunk Common Information Model (CIM) add-on** The CIM add-on version 4.6 or later is required for CIM mapping and creating adaptive response alerts.
  - Download the Splunk Common Information Model add-on from Splunkbase. For details about installation, see Install the Splunk Common Information Model Add-on.
  - The CIM add-on 4.6 or later is also required for end users of your add-on to run adaptive response alerts.

Performance tips

- While developing data inputs, disable unnecessary inputs. Multiple data inputs scheduled to run at frequent intervals consume system resources.
- Running more than 10 add-ons in your development environment might negatively impact Add-on Builder’s performance.

Download and install the Splunk Add-on Builder app

1. Download the Splunk Add-on Builder from Splunkbase.
2. Log into Splunk Web.
3. Click the Manage Apps icon.
4. On the Apps page, click **Install app from file**.
5. Click **Choose File**, navigate to and select the Splunk Add-on Builder package file, then click **Open**.
6. Click **Upload**.

Install the Splunk Add-on Builder app from the command line:

- At the command line, enter:
  ```bash
  splunk install app <path>/<packagefilename>
  ```

- Alternatively, unpack the file and then copy the `/splunk_app_addon-builder` directory to `$SPLUNK_HOME/etc/apps`.

  Ensure that the `/splunk_app_addon-builder` directory and its files have proper permissions and ownership so that Splunk Enterprise can read and write to them.

### Upgrade the Splunk Add-on Builder app

Use Splunk Web to upgrade the Add-on Builder.

Because the Splunk Add-on Builder version 3.0.0 and later is compatible with only Python 3, version 3.0.0 and later of the Add-on Builder is also only compatible with Splunk version 8.0.0, which also runs on Python 3. See the Python 3 Migration manual.

Add-ons built by version 3.0.0 and later that are compatible with Python 2 and Python 3 can run on Splunk version 7.x.

Do not delete the `$SPLUNK_HOME/etc/apps/splunk_app_addon-builder` folder before upgrading the app. You need this folder to configure previous projects in the upgraded version of the Add-on Builder.

1. Download the Splunk Add-on Builder from Splunkbase.
2. Log into Splunk Web.
3. Click Manage Apps.
4. On the Apps page, click **Install app from file**.
5. Click **Choose File** and navigate to the Splunk Add-on Builder package file, then click **Open**.
6. Select **Upgrade app**.
7. Click **Upload**.
8. A pop-up window asks you to migrate to version 3.0.0. Click **Yes**.

### Migrate your existing add-ons

When you migrate your previously created add-ons they become Python 2 and Python 3 compatible. This upgrade cannot be reverted.

The following migration task must be performed using version 8.0.0 of Splunk Enterprise.

1. In your Add-on Builder user interface, click the created add-on you want to upgrade.
2. A pop-up window asks you to upgrade from 2.2.0 to 3.0.0 or later.
3. Click “OK” to upgrade to 3.0.0 or later.
**Migrate modular inputs and alerts**

By default, all modular inputs and alerts built by version 3.0.0 and later are Python 3 compatible on Splunk 8.0, and Python 2 compatible on Splunk 7.x. To run these add-ons with Python 2 on Splunk 8.0, see Documentation:Splunk:8.0.0:Python3Migration:ChangesEnterprise.

**Share data in Splunk Add-on Builder**

The first time you run Add-on Builder, an opt-in message lets you decide whether to send anonymized usage data to Splunk to help improve the app in future releases.

If you opt in, the app enables an internal library to track basic usage and crash information. The library uses browser cookies to track uniqueness and sessions of app users, then sends events to Splunk using XHR in JSON format.

**Data that is not collected**

The following kinds of data are not collected:

- Sensitive data, such as usernames or passwords.
- Identifying information, such as addresses, phone numbers, IP addresses, or hostnames.
- Indexed data in your Splunk platform instance.

**Usage data that is collected**

**Fields common to all events**

The data that the Add-on Builder sends to Splunk includes the following fields that are common to all events.

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Description</th>
<th>Example value</th>
</tr>
</thead>
<tbody>
<tr>
<td>apiKey</td>
<td>string</td>
<td>The MINT API key for the Splunk Add-on Builder.</td>
<td>&quot;4f2f73n&quot;</td>
</tr>
<tr>
<td>appRunningState</td>
<td>string</td>
<td>Not used.</td>
<td>&quot;NA &quot;</td>
</tr>
<tr>
<td>appVersionCode</td>
<td>string</td>
<td>Not used.</td>
<td>&quot;NA&quot;</td>
</tr>
<tr>
<td>appVersionName</td>
<td>string</td>
<td>The version of the Splunk Add-on Builder.</td>
<td>&quot;2.1.0&quot;</td>
</tr>
<tr>
<td>browser</td>
<td>string</td>
<td>The browser name.</td>
<td>&quot;chrome&quot;</td>
</tr>
<tr>
<td>browserVersion</td>
<td>string</td>
<td>The browser version.</td>
<td>&quot;47.0.2526.111&quot;</td>
</tr>
<tr>
<td>carrier</td>
<td>string</td>
<td>Not used.</td>
<td>&quot;NA&quot;</td>
</tr>
<tr>
<td>connection</td>
<td>string</td>
<td>Not used.</td>
<td>&quot;NA&quot;</td>
</tr>
<tr>
<td>device</td>
<td>string</td>
<td>The type of device used.</td>
<td>&quot;MacIntel&quot;</td>
</tr>
<tr>
<td>extraData</td>
<td>object</td>
<td>The version of Splunk Enterprise.</td>
<td>{&quot;splunk_version&quot;: &quot;6.3.2&quot;}</td>
</tr>
<tr>
<td>locale</td>
<td>string</td>
<td>The user locale set in the browser.</td>
<td>&quot;en-US&quot;</td>
</tr>
<tr>
<td>osVersion</td>
<td>string</td>
<td>The version of the operating system.</td>
<td>&quot;OS X 10.11.2&quot;</td>
</tr>
<tr>
<td>packageName</td>
<td>string</td>
<td>The package name of the Splunk Add-on Builder.</td>
<td>&quot;splunk_app-addon-builder&quot;</td>
</tr>
<tr>
<td>platform</td>
<td>string</td>
<td>The platform.</td>
<td>&quot;web&quot;</td>
</tr>
<tr>
<td>remoteIP</td>
<td>string</td>
<td>Not used.</td>
<td>&quot;NA&quot;</td>
</tr>
<tr>
<td>Field</td>
<td>Type</td>
<td>Description</td>
<td>Example value</td>
</tr>
<tr>
<td>------------------</td>
<td>--------</td>
<td>--------------------------</td>
<td>-----------------------</td>
</tr>
<tr>
<td>screenOrientation</td>
<td>string</td>
<td>Not used.</td>
<td>&quot;NA&quot;</td>
</tr>
<tr>
<td>sdkVersion</td>
<td>string</td>
<td>The version of the internal library.</td>
<td>&quot;4.3&quot;</td>
</tr>
<tr>
<td>session_id</td>
<td>string</td>
<td>A unique session identifier.</td>
<td>&quot;a5026251&quot;</td>
</tr>
<tr>
<td>state</td>
<td>string</td>
<td>Indicates whether the browser is online.</td>
<td>&quot;CONNECTED&quot; or &quot;DISCONNECTED&quot;</td>
</tr>
<tr>
<td>userIdentifier</td>
<td>string</td>
<td>Not used.</td>
<td>&quot;NA&quot;</td>
</tr>
<tr>
<td>uuid</td>
<td>UUID</td>
<td>A random identifier that tracks the user uniqueness.</td>
<td>&quot;b1da8edd-6eb5-4620-8fed-6f2e01e2800f&quot;</td>
</tr>
</tbody>
</table>

**Additional fields**

In addition to fields common to all events, the Splunk Add-on Builder sends the following fields to Splunk when usage data is enabled.

<table>
<thead>
<tr>
<th>Event</th>
<th>Source Type</th>
<th>Description</th>
<th>Data (along with common fields)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Session start</td>
<td>mint:ping</td>
<td>Each ping event indicates that a new session has started.</td>
<td><strong>Field</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>fsEncrypted</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>rooted</td>
</tr>
<tr>
<td>Session end</td>
<td>mint:gnip</td>
<td>Each gnip event indicates that a session has ended.</td>
<td>ses_duration</td>
</tr>
<tr>
<td>Page views</td>
<td>mint:view</td>
<td>Triggered once per page view in the app.</td>
<td>current</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>currentView</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>domProcessingTime</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>domLookupTime</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>elapsedTime</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>loadTime</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>previous</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>serverTime</td>
</tr>
<tr>
<td>App performance and</td>
<td>mint:log</td>
<td>Usage and performance logs for the Add-on Builder that track dashboard memory usage, dashboard loading</td>
<td>level</td>
</tr>
</tbody>
</table>
## Example app performance and configuration events

The Add-on Builder sends performance and configuration information using the `log_name` field in the `mint:log` source type. This `log_name` field contains two sub-fields, `name`, which indicates which type of logs are being transmitted, and `data`, the content of the tracking log.

The following examples demonstrate what data the Add-on Builder sends for each type of event.

### Log name

- **track_performance**
  - Tracks dashboard performance for memory usage and loading times.

### Example JSON object

```json
{
  "memory": {
    "totalJSHeapSize": 72200000,
    "usedJSHeapSize": 39600000,
    "jsHeapSizeLimit": 162000000
  },
  "timing": {
    "navigationStart": 1453273923766,
    "unloadEventStart": 1453273923929,
    "unloadEventEnd": 1453273923930,
    "redirectStart": 0,
    "redirectEnd": 0,
    "fetchStart": 1453273923766,
    "domainLookupStart": 1453273923766,
    "domainLookupEnd": 1453273923766,
    "connectStart": 1453273923766,
    "connectEnd": 1453273923766,
    "secureConnectionStart": 0,
    "requestStart": 1453273923773,
    "responseStart": 1453273923927,
    "responseEnd": 1453273923929,
    "domLoading": 1453273923939,
  }
}
```
<table>
<thead>
<tr>
<th>Log name</th>
<th>Example JSON object</th>
</tr>
</thead>
<tbody>
<tr>
<td>track_addon_builder_summary</td>
<td>Tracks information about add-ons created by the Add-on Builder.</td>
</tr>
<tr>
<td></td>
<td>&quot;author&quot;: &quot;splunk&quot;, &quot;id&quot;: &quot;Splunk_TA_test_panda&quot;, &quot;is_validated&quot;: &quot;not validated&quot;, &quot;last_modified&quot;: &quot;2016/10/26&quot;, &quot;name&quot;: &quot;test panda&quot;, &quot;version&quot;: &quot;1.0.0&quot;, &quot;visible&quot;: true</td>
</tr>
<tr>
<td>track_addon_summary</td>
<td>Tracks information about one add-on, including data inputs, alert actions, event types, CIM mappings, and source types.</td>
</tr>
<tr>
<td></td>
<td>{ &quot;source_type&quot;: [{ &quot;data_format&quot;: &quot;unstructured_data&quot; &quot;event_count&quot;: 4584 &quot;has_cim_mapping&quot;: false &quot;has_field_extraction&quot;: false &quot;input_name&quot;: &quot;dsfsdafdsa&quot; &quot;input_type&quot;: &quot;rest&quot; &quot;source_type&quot;: &quot;fdsafdsafdsafdsa&quot; }], &quot;modular_alert&quot;: [{ &quot;description&quot;: &quot;Send HipChat room notifications&quot;, &quot;short_name&quot;: &quot;hipchat&quot;, &quot;label&quot;: &quot;HipChat&quot;, &quot;active_response&quot;: { &quot;group&quot;: [ &quot;information gathering&quot;, &quot;information conve&quot; ] }, &quot;task&quot;: [ &quot;block&quot; ], &quot;technology&quot;: [ { &quot;product&quot;: &quot;endpoint&quot;, &quot;version&quot;: [ &quot;1.0&quot; ] }, &quot;vendor&quot;: &quot;symantec&quot; ] }, &quot;subject&quot;: [ &quot;router&quot;, &quot;firewall&quot; ] }, &quot;alert_props&quot;: { &quot;payload_format&quot;: &quot;json&quot;, &quot;is_custom&quot;: 1 }</td>
</tr>
<tr>
<td>Log name</td>
<td>Example JSON object</td>
</tr>
<tr>
<td>----------</td>
<td>---------------------</td>
</tr>
</tbody>
</table>
| | },
| | "parameters": [ |
| | { |
| | "default_value": "message",
| | "name": "notification_type",
| | "required": true,
| | "help_string": "Choose style of HipChat notification.",
| | "possible_values": { |
| | "Application Card": "card",
| | "Message": "message"
| | },
| | "label": "Notification Style",
| | "format_type": "dropdownlist"
| | },
| | { |
| | "label": "Card Attributes",
| | "required": false,
| | "help_string": "",
| | "name": "card_attributes",
| | "format_type": "text"
| | },
| | { |
| | "label": "Message Format",
| | "required": false,
| | "possible_values": { |
| | "Html": "html",
| | "Plain Text": "plain"
| | },
| | "name": "message_format",
| | "format_type": "radio"
| | },
| | { |
| | "default_value": "red",
| | "name": "color",
| | "required": false,
| | "help_string": "",
| | "possible_values": { |
| | "None": "",
| | "Purple": "purple",
| | "random": "random",
| | "Grey": "grey",
| | "Yellow": "yellow",
| | "Green": "green",
| | "Red": "red"
| | },
| | "label": "Message Color",
| | "format_type": "dropdownlist"
| | },
| | { |
| | "label": "Notify users in the room",
| | "required": false,
| | "name": "notify",
| | "format_type": "checkbox"
| | }
| | }],
| event_type: [{
| "name": "test_event_type" |
| }],
<table>
<thead>
<tr>
<th>Log name</th>
<th>Example JSON object</th>
</tr>
</thead>
<tbody>
<tr>
<td>cim_mapping</td>
<td>cim_mapping:[{&quot;event_type&quot;: &quot;test_eventtype&quot;, &quot;event_field&quot;: &quot;date_year&quot;, &quot;eval&quot;: &quot;date_year&quot;, &quot;cim&quot;: &quot;Alerts&quot;, &quot;cim_field&quot;: &quot;dest&quot;}]</td>
</tr>
<tr>
<td>track_package</td>
<td>{ app_name: 'test addon' }</td>
</tr>
<tr>
<td>track_validation</td>
<td>{ timeStart: 1477547657730, timeEnd: 1477547658745, score: 90, categories: [{'field_extraction', 'app_inspect'}], num_of_warnings: 10, num_of_errors: 0, num_of_passes: 100, app_name: 'ta-test' }</td>
</tr>
<tr>
<td>track_field_extraction</td>
<td>{ timeStart: 1477547657730, timeEnd: 1477547658745, sourcetype: 'test:table', num_of_events: 9933, format: 'unstructured_data', app_name: 'ta-test' }</td>
</tr>
<tr>
<td>track_step_view</td>
<td>{ view: 'field-extraction', page: 'tab_main_flow.html', action: 'edit' }</td>
</tr>
<tr>
<td>track_step_view_wizard</td>
<td>{ view: 'data-collection', wizard_step_name: 'chooseMethod', isEditing: false }</td>
</tr>
<tr>
<td>track_creation</td>
<td>{ type: 'add-on', data: { &quot;friendlyName&quot;: &quot;panda&quot;, &quot;projectAuthor&quot;: &quot;splunk&quot;, &quot;projectDescription&quot;: &quot;a test add-on&quot;, &quot;projectName&quot;: &quot;panda&quot;, &quot;projectNamePrefix&quot;: &quot;Splunk_TA_&quot;, &quot;projectVersion&quot;: &quot;1.0.0&quot;, &quot;themeColor&quot;: &quot;#65A637&quot; } }</td>
</tr>
</tbody>
</table>
"visible": 1

//or
{
  type: 'modular-input'.
  data: {
    "type": "rest",
    "sourcetype": "test:rest",
    "name": "test_rest",
    "title": "test rest",
    "description": "test rest description",
    "interval": "30",
    "data_inputs_options": [{
      "name": "_rest_api_url",
      "rest_header": false,
      "description": "url",
      "value": "http://www.sina.com.cn"
    }, {
      "name": "_rest_api_method",
      "rest_header": false,
      "description": "method",
      "value": "GET"
    }],
    "customized_options": [{
      "name": "string_label",
      "value": "123"
    }],
  }
}

//or
{
  type: 'modular-input'.
  data: {
    "type": "command",
    "sourcetype": "test:command",
    "name": "test_command",
    "title": "test command",
    "description": "this is a test",
    "interval": "30",
    "data_inputs_options": [{
      "name": "_command",
      "description": "command",
      "value": "ls -l"
    }, {
      "type": "customized_var",
      "name": "string_label",
      "title": "Text",
      "description": "",
      "required_on_edit": true,
      "required_on_create": true
    }],
    "customized_options": [{
      "name": "string_label",
      "value": "123"
    }],
  }
}
<table>
<thead>
<tr>
<th>Log name</th>
<th>Example JSON object</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>}</td>
</tr>
</tbody>
</table>
|          |   ","customized_options": [{
|          |     "name": "string_label",
|          |     "value": ""
|          |   }]
<p>|          | }                   |
|          | }                   |
|          | //or }              |
|          |<br />
|          | }                   |
|          | }                   |
|          | //or }              |
|          |<br />
|          | }                   |
|          | }                   |
|          | }                   |
|          | //or }              |
|          |<br />
|          | }                   |
|          | }                   |
|          | }                   |
|          | //or }              |
|          |<br />
|          | }                   |
|          | }                   |
|          | }                   |
|          | //or }              |
|          |<br />
|          | }                   |
|          | }                   |
|          | }                   |
|          | }                   |
|          | }                   |
|          | }                   |
|          | }                   |
|          | }                   |
|          | }                   |
|          | }                   |
|          | }                   |
|          | }                   |</p>
<table>
<thead>
<tr>
<th>Log name</th>
<th>Example JSON object</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>&quot;event_type&quot;: &quot;test_eventtype&quot;,</td>
</tr>
<tr>
<td></td>
<td>&quot;event_field&quot;: &quot;date_year&quot;,</td>
</tr>
<tr>
<td></td>
<td>&quot;eval&quot;: &quot;date_year&quot;,</td>
</tr>
<tr>
<td></td>
<td>&quot;cim&quot;: &quot;Alerts&quot;,</td>
</tr>
<tr>
<td></td>
<td>&quot;cim_field&quot;: &quot;dest&quot;</td>
</tr>
<tr>
<td></td>
<td>}</td>
</tr>
<tr>
<td></td>
<td>//or</td>
</tr>
<tr>
<td></td>
<td>{</td>
</tr>
<tr>
<td></td>
<td>&quot;type&quot;: &quot;modular-alert&quot;,</td>
</tr>
<tr>
<td></td>
<td>&quot;data&quot;: {</td>
</tr>
<tr>
<td></td>
<td>&quot;parameters&quot;: [{</td>
</tr>
<tr>
<td></td>
<td>&quot;required&quot;: false,</td>
</tr>
<tr>
<td></td>
<td>&quot;default_value&quot;: &quot;,&quot;,</td>
</tr>
<tr>
<td></td>
<td>&quot;name&quot;: &quot;password&quot;,</td>
</tr>
<tr>
<td></td>
<td>&quot;label&quot;: &quot;Password&quot;,</td>
</tr>
<tr>
<td></td>
<td>&quot;help_string&quot;: &quot;,&quot;,</td>
</tr>
<tr>
<td></td>
<td>&quot;format_type&quot;: &quot;password&quot;</td>
</tr>
<tr>
<td></td>
<td>}],</td>
</tr>
<tr>
<td></td>
<td>&quot;short_name&quot;: &quot;test_alert&quot;,</td>
</tr>
<tr>
<td></td>
<td>&quot;label&quot;: &quot;test alert&quot;</td>
</tr>
<tr>
<td></td>
<td>}</td>
</tr>
<tr>
<td></td>
<td>}</td>
</tr>
<tr>
<td>track_deletion</td>
<td>Tracks deletion of add-on items.</td>
</tr>
<tr>
<td></td>
<td>{</td>
</tr>
<tr>
<td></td>
<td>type: 'add-on',</td>
</tr>
<tr>
<td></td>
<td>data: {</td>
</tr>
<tr>
<td></td>
<td>&quot;app_name&quot;: &quot;Splunk_TA_panda&quot;</td>
</tr>
<tr>
<td></td>
<td>}</td>
</tr>
<tr>
<td></td>
<td>//or</td>
</tr>
<tr>
<td></td>
<td>{</td>
</tr>
<tr>
<td></td>
<td>type: 'modular-input'.</td>
</tr>
<tr>
<td></td>
<td>data: {</td>
</tr>
<tr>
<td></td>
<td>&quot;type&quot;: &quot;customized&quot;,</td>
</tr>
<tr>
<td></td>
<td>&quot;name&quot;: &quot;test_code&quot;</td>
</tr>
<tr>
<td></td>
<td>}</td>
</tr>
<tr>
<td></td>
<td>//or</td>
</tr>
<tr>
<td></td>
<td>{</td>
</tr>
<tr>
<td></td>
<td>&quot;type&quot;: &quot;sourcetype&quot;,</td>
</tr>
<tr>
<td></td>
<td>&quot;data&quot;: {</td>
</tr>
<tr>
<td></td>
<td>&quot;sourcetype&quot;: &quot;test:st:abc&quot;</td>
</tr>
<tr>
<td></td>
<td>}</td>
</tr>
<tr>
<td></td>
<td>//or</td>
</tr>
<tr>
<td></td>
<td>{</td>
</tr>
<tr>
<td></td>
<td>&quot;type&quot;: &quot;cim-mapping&quot;,</td>
</tr>
<tr>
<td></td>
<td>&quot;data&quot;: {</td>
</tr>
<tr>
<td></td>
<td>&quot;event_type&quot;: &quot;test_eventtype&quot;,</td>
</tr>
<tr>
<td></td>
<td>&quot;event_field&quot;: &quot;date_year&quot;,</td>
</tr>
<tr>
<td></td>
<td>&quot;eval&quot;: &quot;date_year&quot;,</td>
</tr>
<tr>
<td></td>
<td>&quot;cim&quot;: &quot;Alerts&quot;,</td>
</tr>
<tr>
<td></td>
<td>&quot;cim_field&quot;: &quot;dest&quot;</td>
</tr>
<tr>
<td></td>
<td>}</td>
</tr>
<tr>
<td></td>
<td>//or</td>
</tr>
<tr>
<td>Log name</td>
<td>Example JSON object</td>
</tr>
<tr>
<td>----------</td>
<td>---------------------</td>
</tr>
<tr>
<td>track_error</td>
<td></td>
</tr>
</tbody>
</table>
Tracks errors that were triggered in the Add-on Builder. 

```
{
  "err_code": '6099',
  "err_args": ['123'],
  "message": 'Cannot create a modular input with name "123". Input name must start with an alphabetic character, following by alphanumeric characters. ' 
}
```

### How to edit your data collection opt-in or opt-out selections

Change your settings at any time on the **Configuration > Make Add-on Builder Better** tab, which is available from the Splunk Add-on Builder home page.

For more information about how Splunk collects and uses data, please refer to the Splunk Privacy Policy.

### Design your add-on

Before you build your add-on:

- Be familiar with your data and understand the data that you want to extract from it.
- Determine the method you will use to gather your data. For example, if you plan to use file monitors, network listeners, or the HTTP Event Collector, you do not need to build a modular input and can skip the input options requirement.
- If you plan to create a modular input, have sample data and/or a test account for the system that the module will contact. Know the input options that are required to access your data.
- Know which parts of the Common Information Model (CIM) to which you want to map data. For example, almost all data sources produce Authentication and Change Analysis events, but few produce Intrusion Detection events.

### Understanding modular inputs

Modular inputs help you get your data into the Splunk platform when native Splunk data collection inputs do not meet your needs. They are typically written in Python or Java.

For example, use a modular input:

- to query a third-party API for your data
- if you have a type of data that is not a native Splunk input (such as HEC, syslog, or file monitor)

The Splunk Add-on Builder lets you create simple modular inputs for you and generates Python code for those modular inputs. You can also write your own Python code for the data input and provide the input arguments. You can validate each data input by running the input in the Splunk Add-on Builder UI.

For more information, see Modular inputs overview in the Splunk Enterprise *Developing Views and Apps for Splunk Web* manual.
Understanding field extractions

When Splunk Enterprise indexes data, it parses the data stream into a series of events and adds a number of fields to the event data. These fields include:

• default fields that are added automatically
• any custom fields that you specify

The Splunk Add-on Builder lets you add custom fields to your data and perform field mapping at index and/or search time.

For more information, see:

• Create custom fields at index time in the Splunk Enterprise Getting Data In manual.
• Extract fields from files with structured data in the Splunk Enterprise Getting Data In manual

Understanding the CIM

Splunk Add-on Builder lets you normalize data from different sources or vendors to match with a common information model (CIM).

The CIM has 22 predefined data models. To make your data match these standards you add data model mapping to your data. The mappings are applied at search time.

For more information about the CIM, see Overview of the Splunk Common Information Model in the Common Information Model Add-on Manual.

About alert actions

Alerts monitor for and respond to specific events, in real time or on a schedule. When conditions are met, alerts trigger "alert actions". Splunk Enterprise includes a number of pre-configured alert actions, such as running a script, sending an email, or logging an event in response to an alert.

In the Add-on Builder, you create your own alert actions and make them available to your add-on users.

For more information, see

• Custom alert actions overview in the Developing Views and Apps for Splunk Web manual.
• Use the common action model to build custom alert actions in the Common Information Model Add-on Manual
• Create an adaptive response action on the Splunk Developer Portal
Use the Add-on Builder

Use the Splunk Add-on Builder

The Splunk Add-on Builder home page displays:

- projects for add-ons created using the Splunk Add-on Builder
- apps and add-ons that are already installed on your instance of Splunk Enterprise.

Add-on Builder project and meta data is editable and you can export projects to use on other instances of Add-on Builder.

Work with other apps and add-ons

You can work with apps and add-ons that are already installed on your instance of Splunk Enterprise, including apps and add-on that are not created using the Add-on Builder.

- **Validate & Package:**
  - Validate the app or add-on against best practices and other rules and determine whether your app is ready for Splunk App Certification.
  - Download a package file for the app or add-on.
- **Manage Source Type:** Create or import a source type and add sample data for the app or add-on.
- **Extract Fields:** Parse the data in your source types to create field extractions for the app or add-on.
- **Map to Data Model:** Map fields from the app or add-on to the data models.
- **Create Alert Actions:** Create and configure Alert actions for the app or add-on.

If an existing app or add-on has configuration files in the app's `/default` directory, Add-on Builder prompts you to let Add-on Builder move these configuration files to the app's `/local` directory and merge them with any existing configuration files. Typically these configuration files include `props.conf`, `eventtypes.conf`, and `tags.conf`, which are used for data model mapping.

Use Add-on Builder to work with other apps and add-ons

1. On the Splunk Add-on Builder home page, click the Other apps and add-ons tab.
2. On the app or add-on, click the link for the activity you want: Validate & Package, Manage Source Type, Manage Source Type, Extract Fields, Map to Data Model or Create Alert Actions.

Import and export an add-on project

Each add-on you create in the Splunk Add-on Builder has a corresponding project that contains the configuration settings. You must have access to this project in order to modify certain parts of your add-on, including data inputs or field extractions.

Export the add-on project as a TGZ file in order to import the project file to a different instance of Splunk Enterprise running Add-on Builder. This lets you share your add-on project or modify your add-on using a different computer than the one you used to develop the add on.
You can only import and export projects for add-ons that were created in Splunk Add-on Builder.

There are two ways to import and export add-on project: Splunk Web or a migration tool provided by the Splunk Add-on Builder.

- Splunk Web provides a simple process and the recommended path for migrating most add-ons. The exported file is gzipped.
- The migration tool provided by Splunk Add-on Builder lets you automate the process or for code repository check-in. The exported file is an extracted folder.

**Import an add-on project using Splunk Web**

1. On the Splunk Add-on Builder home page on the Created with Add-on Builder tab, click Import Project.
2. Navigate to the add-on TGZ project file and click Open.

**Export an add-on project using Splunk Web**

1. Open the Splunk Add-on Builder home page on the Created with Add-on Builder tab.
2. select the add-on and click Export.

**Import or export a project using the migration tool**

Run the shell command migration tool `project_migration_tool` under

```
$SPLUNK_HOME/etc/apps/splunk_app_addon-builder/bin/aob/aob_tools.
```

Arguments:

```
-h, --help   Show the help of the migration tool and then exit
-l, --log_level {DEBUG,INFO,WARNING,ERROR}  Log level of the migration tool. Default:INFO
-o, --output_dir The output directory to export the project to.
-t, --ta_name The project name of which you want to import/export.
-u, --user Splunk user name. Default:admin
-p, --password Splunk password. Default:changeme
-s, --splunk_web Splunk platform URL. Default: http://localhost:8000
-a, --action {export,import} The operation you want to do. Default:export
-i, --input_directory The source directory of the project.
-v, --version Show the version of the migration tool.
```

**Example One**

1. Import the ServiceNow add-on on github to Splunk add-on builder under $YOUR_FOLDER/ta_servicenow.
2. Run the following command of the migration tool:

```
./project_migration_tool -a import -i $YOUR_FOLDER/ta_servicenow/TA-ServiceNow -s http://localhost:8000 -u admin -p changeme
```
Example two

Export the ServiceNow add-on on your local Splunk instance to github.

1. Run the following command:
   
   ```bash
   ./project_migration_tool -a export -t TA-ServiceNow -o /tmp/ta_servicenow/ -s http://localhost:8000 -u admin -p changeme
   ```
2. Copy the exported content to git repo directory.
   
   ```bash
   cp -r /tmp/ta_servicenow/TA-ServiceNow/* $YOUR_FOLDER/ta_servicenow/TA-ServiceNow
   ```
3. Check in the changes.

Create an add-on

This topic provides an overview of an add-on creation workflow and describes the steps to create a work-flow in the UI.

Add-on Builder workflow

The following diagram shows a common workflow for creating an add-on.
<table>
<thead>
<tr>
<th>Workflow action</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Create a data input</td>
<td>Get data into your add-on by configuring data inputs.</td>
</tr>
<tr>
<td>Configure data collection</td>
<td>Configure a data input by:</td>
</tr>
<tr>
<td></td>
<td>• Using a REST API call</td>
</tr>
<tr>
<td></td>
<td>• Using a shell command</td>
</tr>
<tr>
<td></td>
<td>• writing Python code.</td>
</tr>
<tr>
<td>Create a setup page</td>
<td>Create a setup page for your add-on. A setup page prompt users for information to run your add-on, including authentication.</td>
</tr>
<tr>
<td>Manage source types</td>
<td>Add or import a source type. Add sample data to your add-on so that you can add knowledge objects from it. You can import data from an existing source type or upload sample files.</td>
</tr>
<tr>
<td>Extract fields</td>
<td>Create field extractions from your configured data inputs.</td>
</tr>
<tr>
<td>Map to data model</td>
<td>Map the fields from the field extractions to the data model you want to use.</td>
</tr>
<tr>
<td>Create alert actions</td>
<td>Create custom alert actions, for example, running an adaptive response action for Splunk Enterprise Security.</td>
</tr>
<tr>
<td>Validate the add-on</td>
<td>Validate the add-on for best practices, and test the modular/scripted inputs, field extractions, and data model mappings. The Splunk Add-on Builder shows you any errors or warnings, along with recommendations about how to address them.</td>
</tr>
<tr>
<td>Package the add-on</td>
<td>Package the add-on when you are ready to install your add-on and use it. The result is a compressed file that contains a directory with the necessary configuration files.</td>
</tr>
</tbody>
</table>
Create an add-on

1. Click **Create an add-on** in the Add-on Builder home page.

2. Set the following properties for your project:

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add-on Name</td>
<td>The name of the add-on.</td>
</tr>
<tr>
<td>Author</td>
<td>The author name, which appears with the add-on in Splunk Web.</td>
</tr>
<tr>
<td>Add-on Folder Name</td>
<td>The folder name for the add-on under $SPLUNK_HOME/etc/apps/. Add-on folder names have a prefix of &quot;TA-&quot;. For add-ons created by Splunk, the folder name prefix is &quot;Splunk_TA_&quot;. These prefixes cannot be changed. To change the folder name, click <strong>Edit</strong>.</td>
</tr>
<tr>
<td>Version</td>
<td>The version number of the add-on, which appears with the add-on in Splunk Web.</td>
</tr>
<tr>
<td>Description</td>
<td>The description of your add-on.</td>
</tr>
<tr>
<td>Visible</td>
<td>Indicate whether you want the add-on to be visible in the Splunk Web app list.</td>
</tr>
<tr>
<td></td>
<td>♦ Add-ons are not visible by default unless you create a setup page for your add-on.</td>
</tr>
<tr>
<td></td>
<td>♦ If you create a setup page, the <strong>Visible</strong> property is enabled by default.</td>
</tr>
<tr>
<td>Icon</td>
<td>Upload an image file to use as the icon for your add-on. The Add-on Builder creates a large version (72x72 pixels) and a small version (36x36 pixels) of the icon from this image.</td>
</tr>
<tr>
<td>Theme Color</td>
<td>The color to use for your add-on.</td>
</tr>
</tbody>
</table>

3. Click **Create** to save your settings. The Splunk Add-on Builder displays a home page for your new add-on with the workflow for building it, including a summary of the objects you have configured.

Configure data collection using a REST API call

Use the **Configure Data Collection** page to configure the data inputs and **setup page** for your add-on. The **Configure Data Collection** page provides a list of data inputs that are configured for your add-on. Data inputs determine the method used to get data into your add-on:

- a data input using a REST API call
- A REST API call
- A shell command script
- Modular input from your own Python code

Once you have configured data inputs for your add-on, you can manage them from the **Configure Data Collection** page in the Add-on Builder. You data inputs also appear on the add-on **Inputs** page.

Data inputs must be enabled for Splunk Enterprise to indexing their data. Indexed data is required in order to perform field extractions and data model mapping. Data inputs that you configure are enabled by default.

Your add-on users create data inputs within your add-on. When users open your add-on, the **Inputs** page provides the option to create a new input. Do not direct users to use the **Settings > Data inputs** page in Splunk Enterprise.
Add a data input using a REST API

Build the data collection for your add-on to gather data from a REST API. A REST data input uses JSON as a data type and supports basic authentication and API-based authentication. For advanced data collection, create a modular input by writing your own Python code.

1. On your add-on homepage, click the Configure Data Collection icon.
2. On the Configure Data Collection page, click New Input.
   The Create Data Input wizard starts.
3. On the Choose Input Method page, click Modular input using a REST API.
4. On the Data Input Properties tab:
   ♦ Enter a name for the source type to create for this data input.
   ♦ Enter an input display name, internal name, and description.
   ♦ Specify a collection interval indicating how often to retrieve data using this input, in seconds.
5. Optionally click the Data Input Parameters tab to define any parameters that are required for users to configure your data input. For each parameter, drag a field from the Component Library to the center panel, then specify properties in the Property Editor.
6. Optionally, click the Add-on Setup Parameters tab to define any parameters that are required for users to configure your add-on.
   ♦ Select predefined options to prompt for account, proxy, or logging settings.
   ♦ Add custom inputs by dragging fields from the Component Library to the center panel. Select an input to specify its properties in the Property Editor.

For more about setup pages, see Create a setup page.

7. Click Next.
8. On the Data Input Definition tab under REST settings:
   ♦ Specify a REST URL for a REST endpoint.
   ♦ Select the REST method (GET or POST).
   ♦ If you are using the GET method, specify the names and values for the REST URL parameters.
     You can add parameters individually to the REST header or body, or add them directly to the URL. For details, see Pass values from data input parameters and Pass values from setup parameters.
   ♦ Optionally specify the names and values for REST request headers.
   ♦ If you are using the POST method, optionally specify the names and values for the REST request body.
   ♦ If you added a Global Account field to your data input parameters, optionally select Enable basic authentication to include the username and password values in the HTTP header using HTTP Basic Authentication.
9. On the Data Input Definition tab under Data input parameters, enter a sample value for each data input parameter for testing purposes.
   To use a value from a data input parameter in the REST URL, use the format ${internal_name}, using the internal name defined for the parameter in the previous step. For details, see Pass values from data input parameters and Pass values from setup parameters.
10. On the Data Input Definition tab under Event extraction settings, optionally specify how to break the JSON payload into individual events.
     Use this setting when the results from the API return a JSON array of events, and you would like to break them into individual events before indexing them. Specify the JSON path to that array within the payload to use for event breaking. When the JSON path matches the array in the payload, the extracted events are highlighted in the test output. For more, see JSON path formats.
11. On the Data Input Definition tab under Checkpoint settings, optionally select Enable checkpointing to set a checkpoint.
    A checkpoint allows you to read and index only the most recent data received (data received after the last
most recent data receipt). For more about checkpoints, including a sample configuration, see Use checkpoints for incremental data collection.

Specify the checkpoint settings:

- For **Checkpoint parameter name**, enter the REST parameter name to use as a checkpoint. Make sure you use the `${param_name}` format to replace the checkpoint value in the REST URL definition.
- For **Checkpoint field path**, enter the JSON path to the field in the response payload to use as the checkpoint. For more about specifying the checkpoint field path, see JSON path formats.
- For **Checkpoint initial value**, enter the initial value to use for the checkpoint. Users can modify this value when creating a data input.
- If the timestamp format of the request is different from the response, specify values for **Response timestamp format** and **Request timestamp format**. If both timestamps use the same format, leave these fields empty. For more, see Normalize timestamp formats.

12. If you created a setup page, click the **Add-on Setup Parameters** tab to enter sample values for setup parameters for testing purposes.
13. Click the **Data Input Definition** tab, then click **Test** to run the REST command and test the data input configuration.
   The **Output** section indicates whether the REST command succeeded (sample data is displayed) or failed (the error message is displayed).
14. Click **Save** to save the data input, then click **Finish**.

The Add-on Builder displays a summary of the new data input, along with suggested next steps.

**Add authentication to a data input**

When you need to use authentication for a REST input, add a **Global Setup** input field to your data input parameters form:

The **Global account settings** option is automatically selected on your setup page.

When users first open your add-on, they set up their accounts on the add-on's **Configuration** page:

Then, users configure new inputs on the add-on's **Inputs** page, which displays the data input parameters you configured:

**Use checkpoints for incremental data collection**

When you create your REST API data input, use checkpoints to mark where data has already been read and indexed so that you can resume reading data the next time it is collected. The checkpoint field is typically a timestamp, but you can specify any field. You do not need to use checkpoints with a REST API that returns a point-in-time value, such as metrics.

When using checkpoints, the REST API should do the following:

- Allow you to sort fields by the one you want to use for checkpoints to allow that field to indicate where new data begins.
- Query data by the checkpoint field, starting with the initial checkpoint value.
- Allow timestamps in increments at least as small as the collection interval.
Pass values from data input parameters

You can pass parameter values to a variable in a REST URL, a REST header, a REST body, and in a shell command. Following the Mako syntax, use the format `${varname}` for the variable. Then, define a modular input variable for `varname`.

Pass values from setup parameters

When referring to setup parameters, include a namespace string before the parameter name on order to get the value from the parameter. The following shows an example, where `param_name` is the parameter name:

`${__settings__.additional_parameters.param_name}`

For example, let’s say you have a text input on your setup page that prompts the user for an API token, with the internal name "api_token". To pass the value of the API token to the REST call, use the following format:

`api-key=${__settings__.additional_parameters.api_token}`

We recommend you do not set account, proxy, or logging fields directly. You can access the global account values as follows:

`${global_account.username}`
`{{global_account.username}}`

`${global_account.password}`
`{{global_account.password}}`

JSON path formats

Use the following guidelines to form JSON paths when creating REST inputs.

<table>
<thead>
<tr>
<th>Operator</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$</td>
<td>The root element to query.</td>
</tr>
<tr>
<td>*</td>
<td>Wildcard. Available anywhere a name or number is required.</td>
</tr>
<tr>
<td>..</td>
<td>Deep scan. Available anywhere a name is required.</td>
</tr>
<tr>
<td>.&lt;name&gt;</td>
<td>Dot-notated child.</td>
</tr>
<tr>
<td>[&quot;&lt;name&gt;&quot;]</td>
<td>Bracket-notated child.</td>
</tr>
<tr>
<td>[number]</td>
<td>Array index or indexes.</td>
</tr>
</tbody>
</table>

Example

Here is an example JSON response:

```json
{
    "count": 251,
    "results": [
        {
            "isocode": "MZ",
            "country": "Mozambique"
        }
    ]
}
```
The following table shows different ways of accessing data in the JSON response.

<table>
<thead>
<tr>
<th>JSON path</th>
<th>Returned events</th>
</tr>
</thead>
<tbody>
<tr>
<td>$.count</td>
<td>251</td>
</tr>
<tr>
<td>$.results[*] or $.results</td>
<td>(&quot;isocode&quot;: &quot;MZ&quot;, &quot;country&quot;: &quot;Mozambique&quot;)</td>
</tr>
<tr>
<td></td>
<td>(&quot;isocode&quot;: &quot;PG&quot;, &quot;country&quot;: &quot;Papua New Guinea&quot;)</td>
</tr>
<tr>
<td>$.results[0]</td>
<td>(&quot;isocode&quot;: &quot;MZ&quot;, &quot;country&quot;: &quot;Mozambique&quot;)</td>
</tr>
<tr>
<td>$.results[-1]</td>
<td>(&quot;isocode&quot;: &quot;PG&quot;, &quot;country&quot;: &quot;Papua New Guinea&quot;)</td>
</tr>
<tr>
<td>$.results[0].country</td>
<td>Mozambique</td>
</tr>
<tr>
<td>['$results'][0]['isocode']</td>
<td>MZ</td>
</tr>
<tr>
<td>$.results[0].*</td>
<td>MZ</td>
</tr>
<tr>
<td></td>
<td>Mozambique</td>
</tr>
<tr>
<td>$...isocode</td>
<td>MZ</td>
</tr>
<tr>
<td></td>
<td>PG</td>
</tr>
</tbody>
</table>

When the JSON path matches a field in the payload, the extracted value is highlighted in the test output.
Normalize timestamp formats

When you set a checkpoint, you only need to specify timestamp formats if the format of the request timestamp is different from the response timestamp. In this case, use the following format for timestamps:

\%Y-\%m-\%dT\%H:\%M:\%S.\%f+08:00

Additional notes:

- "0" left padding is added to months, days, hours, minutes, and seconds. To remove left padding, use "-". For example: \%-m.
- "0" right padding is added to milliseconds.
- Add-on Builder supports the time format as described in the Python function time_str2str for transforming a time string to a different format.
- The \%Nf format is supported for to_format only.

Learn more

For more information, see the following Splunk Enterprise documentation:

- How timestamp assignment works in the Getting Data In manual
- Source type naming conventions in the Splunk Add-ons manual
- Modular inputs overview in the Developing Views and Apps for Splunk Web manual
- How to create modular inputs using Splunk SDK for Python on the Splunk Developer Portal
- Create a setup page for a Splunk app on the Splunk Developer Portal

Configure data collection using a shell command

Configure the data inputs and setup page for your add-on in the Configure Data Collection page. This page displays a list of data inputs that configured for your add-on. Data inputs determine the method you use get data into your add-on:

- Use a REST API call
- Run a shell command script
- Create a modular input from your own Python code

After you configure data inputs for your add-on, you can manage them using the Configure Data Collection page in Add-on Builder. Data inputs you configure for your add-on also appear on the Inputs page for that add-on.

- Data inputs must be enabled so that Splunk Enterprise can index their data. Indexed data lets you perform field extractions and CIM mapping. The data inputs you configure are enabled by default.
- For each data input you configure, the Add-on Builder creates an input called "TAB_example", which is visible on the Inputs page in the add-on. This input is used by the Add-on Builder and must not be modified. To change settings for the data input, use the Configure Data Collection page in Add-on Builder.

Add-on users create new create a new input using the Inputs page in that add-on. Do not direct users to use the Settings > Data inputs page in Splunk Enterprise.
Add a data input using shell commands

Get data into your add-on by using a script of shell commands:

1. On your add-on homepage, click Configure Data Collection.
2. On the Configure Data Collection page, click Add Data.
3. On the Choose Input Method page, select Modular input using shell commands.
4. On the Data Input Properties tab:
   - Enter a name for the source type to create for this data input.
   - Enter an input display name, internal name, and description.
   - Specify a data collection interval, in seconds.
5. Optionally, click the Data Input Parameters tab to define parameters that you want to require for users that configure your data input.
   For each parameter, drag a field from the Component Library to the center panel, then specify its properties in the Property Editor.
6. Optionally, click the Add-on Setup Parameters tab to define any parameters that are required for users to configure your add-on.
   - Select predefined options to prompt for account, proxy, or logging settings.
   - Add custom inputs by dragging fields from the Component Library to the center panel panel. Select an input to specify its properties in the Property Editor.
   
   For more about setup pages, see Create a setup page.
7. Click Next.
8. On the Data Input Definition tab:
   - Under Command settings, enter the shell commands.
   - Under Data input parameters, enter a sample value for each data input parameter for testing purposes.

To use a value from a data input parameter in the REST URL, use the format ${internal_name}, using the internal name defined for the parameter in the previous step. For more about passing parameter values to a REST call, see Pass values from data input parameters.
9. If you created a setup page, click the Add-on Setup Parameters tab to enter sample values for setup parameters for testing purposes.
10. Click the Data Input Definition tab, then click Test to run the shell commands and test the data input configuration.
    The Output section indicates whether the shell commands succeeded (sample data is displayed) or failed (the error message is displayed).
11. Click Save to save the data input, then click Finish.

Pass values from data input parameters

Pass parameter values to a variable in a REST URL, a REST header, a REST body, and in a shell command. Following the Mako syntax, use the format $(varname) for the variable. Then, define a modular input variable for varname.

Learn more

For more information, see the following Splunk Enterprise documentation:

- Source type naming conventions in the Splunk Add-ons manual.
- Modular inputs overview in the Developing Views and Apps for Splunk Web manual
• How to create modular inputs using Splunk SDK for Python on the Splunk Developer Portal
• Create a setup page for a Splunk app on the Splunk Developer Portal

Configure data collection using your Python code

In the Configure Data Collection page, configure the data inputs and setup page for your add-on. Data inputs determine how you will get data into your add-on:

• Use a REST API call
• Run a shell command script
• Create a modular input from your own Python code

After you configure data inputs for your add-on, you can manage them from the Configure Data Collection page in Add-on Builder. This page displays a list of data inputs that you have configured for your add-on. Data inputs you configure for your add-on also appear on the Inputs page of the add-on.

• Data inputs must be enabled so that Splunk Enterprise can start indexing the data from them. You'll need indexed data if you want to perform field extractions and CIM mapping. Data inputs you configure are enabled by default.
• For each data input you configure, the Add-on Builder creates an input called "TAB_example", which is visible on the Inputs page of the add-on. Do not modify this input, which is used by the Add-on Builder. Instead, if you want to change settings for the data input, access the options from the Configure Data Collection page in Add-on Builder.

Your add-on's users must create data inputs within the add-on. Do not direct users to use the Settings > Data inputs page in Splunk Enterprise. When users open your add-on, the Inputs page provides the option to create a new input.

Add a data input by writing your own code

You can implement your own data collection mechanism using Python code.

Apps created in the Splunk Add-on Builder version 3.0.x, are only compatible with Splunk version 8.x. Add-ons built in version 3.0.x that are built to be compatible with Python 2 and Python 3 can run on Splunk version 7.x. See the Python 3 Migration manual

1. On your add-on's homepage, click the Configure Data Collection icon.
2. On the Configure Data Collection page, click Add Data to start the Create Data Input wizard.
3. On the Choose Input Method page, select Modular input using your own Python code.
4. On the Data Input Properties tab:
   ♦ Enter a name for the source type to create for this data input.
   ♦ Enter an input display name, internal name, and description.
   ♦ Specify a collection interval, in seconds, to indicate how often the add-on should retrieve data using this input. Do not set a collection interval for a single-instance input.
5. Click the Data Input Parameters tab to define any parameters that are required for users to configure your data input. At least one parameter is required.
   For each parameter, drag a field from the Component Library to the center panel, then specify its properties in the Property Editor.
6. Optionally, click the Add-on Setup Parameters tab to define any parameters that are required for users to configure your add-on.
   ♦ Select predefined options to prompt for account, proxy, or logging settings.
Add custom inputs by dragging fields from the Component Library to the center panel. Select an input to specify its properties in the Property Editor.

For more about setup pages, see Create a setup page.

7. Click Next.
8. On the Data Input Definition tab in the Code Editor panel, edit the Python code to create your modular input.
   ♦ Add code under the `validate_input` function to provide validation logic.
   ♦ Add code under the `collect_events` function to add data collection logic.
   ♦ Use the Python helper functions, which are dynamically generated as commented code, as a guide to working with the input parameters you defined in the previous step.
   The autogenerated code contains a working example that generates a random number.
   ♦ Under Data input parameters, enter sample values for testing the data input.

9. If you created a setup page, click the Add-on Setup Parameters tab to enter sample values for setup parameters for testing purposes.
10. Click the Data Input Definition tab, then click Test to run your code and test the data input configuration. The Output section indicates whether your code succeeded (sample data is displayed) or failed (the error message is displayed).
11. Click Save to save the data input, then click Finish.

The Add-on Builder displays a summary of the new data input, along with suggested next steps.

Add authentication to a data input

To use authentication for a REST input, add a Global Setup input field to your data input parameters form.

The Global account settings option is automatically selected on your setup page. To access a username and password in your code, use the following format:

```python
import json
global_account = helper.get_arg('global_account')
username = global_account['username']
password= global_account['password']
```
When users first open your add-on, they set up their accounts on the add-on's **Configuration** page. Then they configure new inputs on the add-on's **Inputs** page, which displays the data input parameters you configured:

### Pass values from data input parameters

Add-on Builder generates the Python helper functions for you to access the values from data input parameters, where "server" is the global parameter name, and "local" is the modular input parameter name:

```python
server = helper.get_global_setting("server")
local = helper.get_param("local")
```

For more, see [Python helper functions](#).

### Single-instance and multi-instance modes

Multi-instance mode runs one instance of a data input per input stanza, resulting in multiple instances (one for each input stanza). This default behavior is useful in multi-thread environments or in situations that require different security contexts or access to different databases.

In a single-thread environment you can run an input in single-instance mode. For example, in a WMI environment you would run a single instance of an input to re-use connections.

Multi-instance mode is enabled by default. To enable single-instance mode, do not set a collection interval for the data input, and then uncomment the following lines in the Python code for the data input:

```python
def use_single_instance_mode():
    return True
```

### Learn more

For more information, see the following Splunk Enterprise documentation:

- Source type naming conventions in the *Splunk Add-ons* manual.
- Modular inputs overview in the *Developing Views and Apps for Splunk Web* manual
- How to create modular inputs using Splunk SDK for Python on the Splunk Developer Portal
- Create a setup page for a Splunk SDK for Python on the Splunk Developer Portal

### Create a setup page

Creating a setup page for your add-on lets users provide information needed to run your add-on without having to edit a configuration file directly. Create a setup page that appears to users when they run your add-on. Use a setup page when:

- Your add-on requires setup parameters.
- You need to authenticate users with a username and password, for example to authenticate a third-party API for a data input.
- You need the user to provide proxy information, for example when you need to use a proxy to make outbound API calls from the add-on.
Add a setup page to your add-on

The Add-on Builder provides a wizard that walks you through the process of creating a setup page and includes the following steps:

Create the setup page

The Add-on Builder wizard includes an **Add-on Setup Parameters** tab where you create or edit the setup page for an add-on. Select predefined setup options and define custom text, password, and checkbox inputs. Setup parameters can be used in any data input or alert action within the add-on.

You can use the following setup parameters when you create your setup page.

<table>
<thead>
<tr>
<th>Setup parameter</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Global account settings</td>
<td>Adds a predefined <strong>Account</strong> tab to the add-on's <strong>Configuration</strong> page that allows the user to configure multiple accounts, such as one account for production and one for development. This <strong>Global account setting</strong> option is selected automatically when you add a <strong>Global Account</strong> field as a data input parameter.</td>
</tr>
<tr>
<td>Proxy settings</td>
<td>Adds a predefined <strong>Proxy</strong> tab to the add-on's <strong>Configuration</strong> page that allows the user to specify a proxy host, port, username, password, and type.</td>
</tr>
<tr>
<td>Logging settings</td>
<td>Adds a predefined <strong>Logging</strong> tab to the add-on's <strong>Configuration</strong> page that lets the user select the logging level to use for the add-on.</td>
</tr>
<tr>
<td>Text field</td>
<td>Drag and drop different types of inputs from the Component Library to the setup form. These options are displayed on a separate <strong>Add-on Settings</strong> tab on the add-on's <strong>Configuration</strong> page. Password fields are automatically obfuscated using the storage password endpoint.</td>
</tr>
<tr>
<td>Password field</td>
<td></td>
</tr>
<tr>
<td>Checkbox field</td>
<td></td>
</tr>
</tbody>
</table>

Test the setup page

Test the setup page by entering sample values.

View the production version of the setup page

To access the production version of the add-on's setup page while developing your add-on, open the app in Splunk Web and click **Configuration**.

For detailed procedures, see:

- Configure data collection using a REST API call
- Configure data collection using your Python code
- Create an alert action
Add sample data

The Add Sample Data page lets you add sample data to your add-on. Any source types you have already defined for your add-on, such as the data inputs you configured from modular inputs in the Configure Data Collection section, are also listed on this page.

Uploading sample data is useful when:

- You have configured a data collection and you want to upload more sample data to create knowledge objects such as field extractions and to perform CIM mapping.
- Your add-on relies on native core data inputs for data collection (for example, syslog files or the HTTP Event Collector) and you want to use the Add-on Builder to create knowledge objects and perform CIM mapping.

You can add sample data in two ways:

- Upload sample data from one or more files and create a source type for this data input.
- Add data that is already indexed in your Splunk Enterprise instance by selecting an existing source type.

Sample data counts against your license.

Add sample data from a file

After you add data inputs in the Configure Data Collection section, you can use the Add-on Builder to upload sample data files for your source types. This can be helpful when you want to create knowledge objects immediately, instead of waiting for newly created modular inputs to collect necessary data.

You cannot upload compressed files.

To upload a sample data file and create a source type

1. On your add-on homepage, click Add Sample Data on the Add-on Builder navigation bar.
2. On the Add Sample Data page, click Add From File.
3. Enter a source type name for this data.
4. Click Upload Data
   5. Navigate to and select the sample data file, then click Open. The preview displays the first 1000 events from the first 2MB of data.
   6. Adjust indexing settings as needed:
      ♦ Expand the Event Breaks section and select an option that indicates how events for the data in this source type should be separated:
         ◊ Auto: Events are auto-detected based on their timestamp location.
         ◊ Every Line: Every line is one event.
         ◊ Regex: Use a regular expression to define a pattern to split events.
      ♦ Expand the Timestamp section and select an option that indicates how to generate timestamps for the data.
      ♦ Expand the Advanced section to specify additional index-time parameters for parsing data.
   7. Click Save.

Sample events are stored in a dedicated “add_on_builder_index” index.
To upload a sample data file for an existing source type

1. On your add-on homepage, click **Add Sample Data** on the Add-on Builder navigation bar.
2. On the **Add Sample Data** page, find the source type in the table and click **Add Sample**.
3. Click **Upload Data**, navigate to and select the sample data file, then click **Open**.

   The preview displays the first 1000 events from the first 2MB of data.

4. Adjust indexing settings as needed:
   ♦ Expand the **Event Breaks** section and select an option that indicates how events for the data in this source type should be separated:
     ◊ **Auto**: Events are auto-detected based on their timestamp location.
     ◊ **Every Line**: Each line is one event.
     ◊ **Regex**: Use a regular expression to define a pattern to split events.
   ♦ Expand the **Timestamp** section and select an option that indicates how to generate timestamps for the data.
   ♦ Expand the **Advanced** section to specify additional index-time parameters for parsing data.
5. Click **Save**.

Add indexed data from Splunk Enterprise

To add data that has already been indexed in Splunk Enterprise

1. On your add-on homepage, click **Add Sample Data** on the Add-on Builder navigation bar.
2. On the **Add Sample Data** page, click **Add From Splunk**.
3. Select the source type of the data to add.
4. Click **Add**.

Any future changes made to the original source type will not be included in your add-on.

Learn more

For more information, see the following Splunk Enterprise documentation:

- For source types, see Configure source types in the *Getting Data In* manual
- For event breaks, see Configure event line breaking in the *Getting Data In* manual
- For advanced settings, see props.conf in the *Admin Manual*

Manage source types

Source types let you categorize your data for easier searching. To learn about source types, see Why source types matter in the *Getting Data In* manual.

You can add new source types in the Add-on Builder:

- By creating a new source type and uploading sample data from one or more files for this source type.
- By importing an existing source type from the Splunk platform.
Add a new source type

1. On your add-on homepage, click Manage Source Types on the Add-on Builder navigation bar.
2. On the Manage Source Types page, click Add and then New Source Type.
3. Enter a unique source type name.
4. Click Upload Data.
5. Select the sample data file, then click Open.

The preview displays the first 1000 events from the first 2MB of data.

6. Adjust indexing settings as needed:
   - Expand the Event Breaks section and select an option that indicates how events for the data in this source type should be separated:
     - Auto: Events are auto-detected based on their timestamp location.
     - Every Line: Every line is one event.
     - Regex: Use a regular expression to define a pattern to split events.
   - Expand the Timestamp section and select an option that indicates how to generate timestamps for the data.
   - Expand the Advanced section to specify additional index-time parameters for parsing data.
7. Click Save.

Sample events are stored in a dedicated "add_on_builder_index" index.

Import an existing source type

1. On your add-on homepage, click Manage Source Types on the Add-on Builder navigation bar.
2. On the Manage Source Types page, click Add and then Import From Splunk.
3. Select a source type from the drop-down list.
4. (Optional) Click Upload Data, select the sample data file, then click Open.

The preview displays the first 1000 events from the first 2MB of data.

5. Adjust indexing settings as needed:
   - Expand the Event Breaks section and select an option to indicate how events should be separated:
     - Auto: Events are auto-detected based on their timestamp location.
     - Every Line: Every line is one event.
     - Regex: Use a regular expression to define a pattern to split events.
   - Expand the Timestamp section and select an option to indicate how to generate timestamps for the data.
   - Expand the Advanced section to specify additional index-time parameters for parsing data.
6. Click Save.

Edit an existing source type

1. On your add-on homepage, click Manage Source Type on the Add-on Builder navigation bar.
2. Click Edit on the source type you want to edit.
3. (Optional) Click Upload Data, navigate to and select the sample data file, then click Open.

The preview displays the first 1000 events from the first 2MB of data.

4. Adjust indexing settings as needed:
   - Expand the Event Breaks section and select an option that indicates how to separate events:
◊ **Auto**: Events are auto-detected based on their timestamp location.
◊ **Every Line**: Every line is one event.
◊ **Regex**: Use a regular expression to define a pattern to split events.

- Expand the **Timestamp** section and select an option that indicates how to generate timestamps for the data.
- Expand the **Advanced** section to specify additional index-time parameters for parsing data.

**Learn more**

- Configure source types in the *Getting Data In* manual
- Configure event line breaking in the *Getting Data In* manual
- props.conf in the *Admin Manual*

**Extract fields**

Use **Extract Fields** functionality to parse the data in your source types and create field extractions.

**Parse data**

To extract fields from your data, you must parse the data for each of the source types in your add-on. The Field Extractor supports parsing for the following data formats:

- **Unstructured Data.** Typically used for log files.
- **Table.** Data in tabular formats, such as comma-separated values (CSV) and tab-separated values (TSV).
- **Key Value.** Data that contains key-value pairs.
- **JSON.** Data in the JavaScript Object Notation (JSON) format.
- **XML.** Data in the Extensible Markup Language (XML) format.

**To parse data for a source type and extract fields**

1. On your add-on homepage, click **Extract Fields** on the Add-on Builder navigation bar.
2. On the **Extract Fields** page, from **Sourcetype**, select a source type to parse.
3. From **Format**, select the data format of the data. Any detected format type is automatically selected and you can change the format type as needed. If you aren't sure what format type you need and a format type has not been automatically selected, use "Unstructured Data" as the format type.
4. Click **Parse**.

**Extract fields**

After parsing the data, the Add-on Builder displays the results on a summary page.

- If you are satisfied with the results, click **Save**.
- If you want to try parsing again using a different format, click **Cancel** to return to the previous page.

After data for a source type has been parsed, the source type is added to the table on the **Extract Fields** page.

- To retrieve parsed field extractions, click **Load Results** for the source type.
**Unstructured Data**

The Add-on Builder's field extractor displays a selection of events in groups, along with the extracted fields. Use this display to:

- Select one or more groups to represent the data.
- Display the regular expression that the field extractor used, and modify it to improve the field extraction.
- Click on individual field names to include or exclude the field for extraction.
- Click the Edit icon next to a field name to edit the field name.
- Click the Trash icon next to a field name to remove its capture group from the regular expression.

**Table**

The Table format is used with tabular data and lets you:

- Change how data is parsed by selecting the delimiter character that is used to separate fields. To specify a different character, click **Other** and enter the character.
- Change the field names after you have selected the correct delimiter. Note that each time you change delimiters, the number of columns might change and cause you to lose changes to field names.

**Key Value**

The Key Value format is used with data containing key-value pairs and lets you do the following:

- Change how data is parsed. For **Extraction Methods**, you can select:
  - **Auto** to let the Add-on Builder parse data automatically.
  - **Delimiters** to use delimiters.
  - **Regex** to use regular expressions.

- For **Delimiters**, select the delimiters for the key-value pairs:
  - Specify the pair delimiter character, which is used to separate key-value pairs.
    Using the example `key_a=value_a, key_b=value_b`, the correct character is a comma.
  - Specify the key-value delimiter character, which is used to separate keys and values.
    Using the example `key_a=value_a, key_b=value_b`, the correct character is an equals sign.

- For **Regex**: select the regular expression to use, or create your own.

**JSON**

The JSON format is used with JSON data. There are no additional parsing options.

**XML**

The XML format is used with XML data. There are no additional parsing options.

**Troubleshooting**
What if I need to upload different sample data?

If you decide that you need to upload a different sample data file for a source type, for example you want to clean the data first, go to Add Sample Data, delete the sample data, then upload additional data files.

A regular expression had too many capture groups, what do I do?

This error is displayed after attempting to parse a file, and the regular expression created by the Field Extractor contains more than 100 capture groups (fields).

This error might indicate a problem with the Event Break setting for the source type:

1. Go to Add Sample Data.
2. Edit the source type and select a different option for Event Break.
3. Upload the sample events again. Because the Event Break option is applied when indexing the data, changing this value does not affect events that have already been indexed.
4. Parse the data again.

The sample data might contain an event that is too long:

1. Edit the sample data file by splitting the long lines to clean up the data.
2. Go back to Add Sample Data.
3. Upload the sample events again.
4. Parse the data again.

Why are the field names not detected in my tabular data?

The Add-on Builder uses the first 1000 events for field extraction. If your data contains more than 1000 events, the parser cannot automatically detect the field names.

The parser assumes that all entries except the table header contain a timestamp. If entries in your tabular data do not contain a timestamp, the parser will not correctly detect which entry is the table header.

Learn more

For more information, see the following Splunk Enterprise documentation:

- About fields in the Knowledge Manager Manual
- Build field extractions with the field extractor in the Knowledge Manager Manual
- Field Extractor: Select Fields step in the Knowledge Manager Manual

Map to data model

Version 2.2.0 and later of the Splunk Add-on Builder lets you map the fields from your data events to the fields in any data model, including CIM data models.

- If you want to map your data to a CIM data model, the Splunk Common Information Model add-on is required to use this feature. Download the Splunk Common Information Model add-on from Splunkbase and see Install the Splunk Common Information Model Add-on for details on how to install this add-on.
• If you want to map to your own data model, the model needs to support the standard defined under the Create a data model section.

Before you apply the data model mapping to your add-on, you must configure one or more source types for your add-on by creating a data input, by adding data from a sample file, or by adding indexed data from Splunk.

Configure the following.

In Map to data model, map the fields from your data to the fields in one of the predefined data models to normalize data at search time.

1. On your add-on homepage, click Map to data model on the Add-on Builder navigation bar.
3. On the Data Model Mapping >> Define Event Type page, define an event type to generate events from which to extract fields:
   ♦ Enter a name for the event type.
   ♦ Select a source type from which to generate events.
   ♦ Enter a search to select events. By default, the search selects all events for the source type you selected, but you can apply additional search criteria as needed.
   ♦ Click Save.
5. On the Data Model Mapping >> Select Data Models page, select the data model to use for mapping:
   ♦ From the center panel, select one or more data models to use. Then you can also select individual datasets within a data model. Fields from your event type are displayed for reference, and fields from the selected data models are also displayed.
   ♦ When you have finished selecting data models, click Done.
6. In Data Model Mapping Details, click New Knowledge Object and select the type of mapping to create:
   ♦ Select FIELDALIAS to map a field from the data model to a field from your event type.
   ♦ Select EVAL to map a field from the data model to an expression based on a field from your event type.
7. Define a field alias or expression in the new row that was added to the Data Model Mapping List:
   ♦ To define a field alias, click one field name from the Data Model Fields list and one from the Event Type Fields list. Define the field alias, then click OK at the end of the new row in the Data Model Mapping List.
   ♦ If you are defining an expression, click one field name from the Data Model Fields list and one or more fields from the Event Type Fields list. Edit the expression in the Event Type Field or Expression column, then click OK at the end of the new row in the Data Model Mapping List.
8. Repeat steps 6-7 as needed.
9. Click Done when you have finished data model mapping.

The Data Model Mapping page displays an entry for the mapping you just completed.

Learn more

For more information, see the following Splunk Enterprise documentation:

- About event types in the Knowledge Manager Manual
- About tags and aliases in the Knowledge Manager Manual
- Use the CIM to normalize data at search time in the Common Information Model Add-on Manual
Create alert actions

In Create Alert Actions, create and configure alert actions to make them available to your add-on's users. You can use Alert actions to define third-party integrations, add custom functionality, or support adaptive response actions for Splunk Enterprise Security.

After you add alert actions to your add-on, you can manage them from the Alert Action page under Create Alert Actions. Your alert actions also appear on the Settings > Alert Actions page in Splunk Web.

Create an alert action

1. On your add-on homepage, click the Create Alert Actions icon.
   The Create Alert Action wizard starts.
3. On the Alert Action Properties page, enter the properties for this alert action:
   - Enter a name, label, and description for this alert action. The label is the friendly name that appears in Splunk Web.
   - Optionally, click Upload my icon to add an icon, such as a logo, to this alert action.
   - To create an adaptive response action, see Create an adaptive response action for Enterprise Security, below.
4. Click Next.
5. On the Alert Action Inputs tab, specify one or more input fields for this alert action.
   For each input, drag a field from the Component Library to the center panel, then specify its properties in the Property Editor.
   If certain permission is required to trigger the alerts, you can drag Text field and Password field on the Component Library, specify the properties and then add the account in the Add-on Setup Parameters page as the example below.
   Optionally, click the Add-on Setup Parameters tab to define any parameters that are required for users to configure your add-on.
   - Select predefined options to prompt for account, proxy, or logging settings.
   - Add custom inputs by dragging fields from the Component Library to the center panel panel. Select an input to specify its properties in the Property Editor.
   For more about setup pages, see Create a setup page.
6. Optionally, click the Add-on Setup Parameters tab to define any parameters that are required for users to configure your add-on.
7. Click Next.
8. On the Alert Action Parameters tab, use the Code Editor panel to edit the Python code to create your alert action. Find the comment # TODO: Implement your alert action logic here, which indicates where to add your code.
9. The Python helper functions, which are dynamically generated as commented code, can help you as a guide to working with the input parameters you defined in the previous step.
10. If you want to collect data using SSL certificate, use send_http_request function as follows.

```python
import solnlib.splunkenv

cert_file_path = solnlib.splunkenv.make_splunkhome_path(['<TA_folder>', 'default', 'example.pem'])
response = helper.send_http_request('https://www.example.com/api', 'GET', verify=cert_file_path)
```
11. helper.settings is a dict that contains information including SPL, Splunk environment configurations and alert configurations. If you want to use the information in helper.settings, use the following sample code as an example.
Syntax:
search_name = helper.settings.get('search_name')
sid = helper.settings.get('sid')

12. On the **Alert Action Parameters** tab, enter sample values for testing this alert action.
13. If you created a setup page, on the **Add-on Setup Parameters** tab enter sample values to test the setup page.
14. Click **Test** to test your code and the alert action configuration. The **Output** section indicates whether the command succeeded or failed.
15. Click **Save**, then click **Finish**.

**Create an adaptive response action for Enterprise Security**

An adaptive response action is a type of alert action that is specifically created for Splunk Enterprise Security. An adaptive response action can be triggered from correlation searches or on an ad hoc basis when examining a notable event on the Incident Review dashboard.

Use the Splunk Add-on Builder to create an alert action that supports adaptive response. Adaptive response actions must conform to the Common Action Model, which is included with the Splunk Common Information Model add-on.

**Note** The Splunk Common Information Model add-on version 4.6.0 or later is required by:

- Add-on developers to create adaptive response actions
- End users of your add-on to run adaptive response action

Download the Splunk Common Information Model add-on from Splunkbase.

**To create an adaptive response**

1. On your add-on homepage, click the **Create Alert Actions** icon.
2. On the **Alert Actions** page, click **New Alert Action**.
3. On the **Alert Action Properties** page, enter the properties for this alert action:
   - Enter a name, label, and description for this alert action. The label is the friendly name that appears in Splunk Web.
   - Optionally, click **Upload my icon** to add an icon, such as a logo, to the alert action.
4. Select **Support as an adaptive response action in Splunk Enterprise Security** and fill out the fields as follows:
   - For **Category**, enter the categories the action belongs to, such as “Information Gathering”.
   - For **Task**, enter the functions performed by the action, such as "scan".
   - For **Subject**, enter the objects that the action's tasks can be performed on, for example, "endpoint.file".
   - For **Vendor**, enter the technology vendor that the action supports.
   - For **Product**, enter the products that the action supports.
   - For **Version**, enter the versions of the product that the action supports.
5. Select **Support as an ad hoc action** if the action supports ad hoc invocation from the Actions menu on the Incident Review dashboard in Splunk Enterprise Security. For help determining whether your action supports ad hoc invocation, see Determine whether your action supports ad hoc invocation on the Splunk Developer Portal.

Then, fill in the related fields as follows:
- Optionally, for **Custom drilldown**, enter a URL to a custom drilldown or view for the link that appears in the detailed view of a notable even on the Incident Review dashboard in Splunk Enterprise Security. If you don't specify a URL, the default URL runs a search for the result events created by this response action.
To specify a target in an app outside Enterprise Security, use the format
`../<app_context>/<viewname>?<additional drilldown parameters>`.
For example:

`../my_app/my_view?form.sid=$orig_sid$&form.rid=$orig_rid$`

To redirect to a custom view within Enterprise Security, use the format
`/<viewname>?<additional drilldown parameters>`.
For example:

`/my_view?form.sid=$orig_sid$&form.rid=$orig_rid$`

♦ For **Sourcetype**, enter the source type to which to assign the events produced as a result of this response action.

6. Click **Next**.
7. On the **Alert Action Inputs** tab, specify one or more input fields required for configuring the alert action.

   For each input, drag a field from the Component Library to the center panel, then specify its properties in the Property Editor.

8. Optionally, click the **Add-on Setup Parameters** tab to define any parameters that are required for users to configure your add-on.

   ♦ Select predefined options to prompt for account, proxy, or logging settings.
   ♦ Add custom inputs by dragging fields from the Component Library to the center panel panel. Select an input to specify its properties in the Property Editor.

   For more about setup pages, see [Create a setup page](https://docs.splunk.comDocumentation/develdoc/)

9. Click **Next**.
10. On the **Alert Action Parameters** tab, use the Code Editor panel to edit the Python code to create your alert action. Find the comment `# TODO: Implement your alert action logic here`, to see where to add your code.

11. Use the Python helper functions, which are dynamically generated as commented code, as a guide to working with the input parameters you defined in the previous step.

12. For example Python code showing how to create an adaptive response, see [Walkthrough: Create an ES adaptive response action](https://docs.splunk.comDocumentation/develdoc/)

13. On the **Alert Action Parameters** tab, enter sample values for testing the alert action.

14. If you created a setup page, on the **Add-on Setup Parameters** tab, enter sample values to test the setup page.

15. Click **Test** to test your code and the alert action configuration. The **Output** section indicates whether the command succeeded or failed.

16. Click **Save**, then click **Finish**.

You should also test and validate your response action in Enterprise Security. For more, see [Validate your response action in Enterprise Security](https://docs.splunk.comDocumentation/develdoc/)

### Pass values from setup parameters

When referring to setup parameters, include a namespace string before the parameter name to get the value from the parameter. For example, where `param_name` is the parameter name:

```python
${__settings__.additional_parameters.param_name}
```

If you have a text input on your setup page that prompts the user for an API token, with the internal name "api_token", use the following format to pass the value of the API token to the REST call:

```python
api-key=${__settings__.additional_parameters.api_token}
```
Do not set account, proxy, or logging fields directly. You can access the global account values as follows:

${global_account.username}

{{global_account.username}}

${global_account.password}

{{global_account.password}}

Read user credentials from multiple accounts

Alert actions do not support the Global Account input field. When you have alert actions that require different user credentials for different accounts, such as one account for production and one for development, set up your alert actions to read specific user credentials from the setup page.

The following workflow shows one way to read multiple credentials of a setup page.

1. Create an alert action.
2. When creating a setup page on the Add-on Setup Page Parameters tab in the Create Alert Action wizard, select Add Global account settings to prompt users to add credentials for one or more accounts.
3. When defining the input variables for the alert action, add a Text field that prompts for the username of an account. The value of this text field acts as a dictionary key to look up the corresponding credentials from the setup page.
4. When defining the code for the alert action, use the following code to retrieve a username and password from the setup page.
   
   Use the same internal name (replace "internal_name" below) that you used for the text field from the previous step:
   
   ```python
   account = helper.get_user_credential(helper.get_arg("internal_name"))
   ```
   
   The data returned by the helper.get_user_credential function is a JSON dictionary, so retrieve the username and password from the account dictionary as follows:
   
   ```python
   username = account["username"]
   password = account["password"]
   ```
5. Save your alert action, then restart Splunk Enterprise.

For more, see the Python helper functions.

Test your alert action

Use the sendalert command using a hard-coded value to test your alert action:

1. Build your alert action and save it.
2. Restart Splunk Enterprise.
3. Use sendalert command:

   ```bash
   index=_internal | head 1| eval fieldname="xyz" | sendalert myalertname param.abc="myvalue"
   ```

To pass search result values dynamically to different alert action parameters, use the $result.fieldname$ format:

```bash
index=_internal | head 1| eval fieldname="xyz" | sendalert myalertname param.abc="$result.fieldname$"
```
For a list of possible alert action tokens, see Pass search result values to alert action tokens in the *Developing Views and Apps for Splunk Web* manual.

**Learn more**

For more information, see the following documentation:

- For creating adaptive response actions, see the Adaptive Response Framework on the Splunk Developer Portal
- For creating adaptive response actions, see Use the common action model to build custom alert actions in the *Common Information Model Add-on Manual*
- For installing the common action model, see Install the Splunk Common Information Model Add-on in the *Common Information Model Add-on Manual*

**Validate and package**

In *Validate & Package*:

- Validate your add-on for best practices and Splunk App Certification, and test your data inputs, field extractions, and data model mappings. The Splunk Add-on Builder shows you any errors or warnings, along with recommendations about how to address them.
- Download your add-on package as an SPL file. The SPL format is identical to the tar archive format, but uses the .spl extension rather than .tar.gz.

You can also validate and package apps and add-ons that were not created using the Add-on Builder.

**About Splunk App Certification**

Splunk App Certification is a process that analyzes your add-on and certifies that it meets certain criteria and verifies that your add-on conforms to best practices. Criteria and best practices may change, so for the latest information see About app certification on the Splunk Developer Portal.

The Splunk Add-on Builder can validate your add-on to determine whether it is ready to be submitted to Splunk App Certification. When you select *App Precertification* from the types of validation to perform, the Add-on Builder logs into the App Certification server using your Splunk.com credentials, runs the app precertification validation, then displays a "ready" or "not ready" message.

Before you run validation, be sure to provide your Splunk.com credentials.

**To set Splunk.com credentials**

1. From the Splunk Web home page or from the *Apps* menu, click *Splunk Add-on Builder*.
2. On the Splunk Add-on Builder home page, click *Configuration* at the top left of the page.
3. On the Configuration page, enter your Splunk.com username and password.
4. If you use a proxy server, select *Enable proxy* and fill out the proxy settings.
5. To verify your login credentials and proxy settings, click *Test*.
6. Click *Save*.
Validate the add-on

You can select from the following types of validation to perform on your add-on:

- Best Practices
- Data Model Mapping
- Field Extraction
- App Precertification

To validate your add-on

1. On your add-on homepage, click Validate & Package on the Add-on Builder navigation bar.
2. Click Validate to run the validation and display the results.

<table>
<thead>
<tr>
<th>Overall Health Report</th>
<th>This report displays a health validation score based on the errors, warnings, and failed rules. If you are validating for app precertification, this section also lets you know whether your app is ready to be submitted to Splunk App Certification. This health score is valid in a local environment. The score could be different for the same add-on if the add-on were validated in other environments or at different times due to differences in global knowledge management and differences in indexed data. Use the health score as a subjective indicator about the overall quality of your add-on.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Error</td>
<td>The total number of errors that were found.</td>
</tr>
<tr>
<td>Warning</td>
<td>The total number of warnings that were found.</td>
</tr>
<tr>
<td>Pass</td>
<td>The number of rules that your add-on passed.</td>
</tr>
<tr>
<td>Validation result distribution</td>
<td>A chart displaying the breakdown of the validation results.</td>
</tr>
<tr>
<td>Validation Rule Details</td>
<td>A list of all the validation rules, with details about the validation category, a description of the rule, and a suggested solution to fix the problem for rules that your add-on did not pass. Sometimes the suggested remedy is to directly edit a configuration file (.conf). The configuration files for your app are located in $SPLUNK_HOME/etc/apps/TA_your_addon_name, and you can edit them in a text editor. After you save changes to the file, refresh Splunk Web by going to http://<a href="">localhost:port</a>/en-US/debug/refresh.</td>
</tr>
</tbody>
</table>

For more information about the best practices for add-ons, see:

- Add-on Best Practice Check Tool

Troubleshoot validation for app precertification

If the validation process hangs during app precertification, the App Certification server might not be responding.

To troubleshoot the source of the problem, run the following search command:

```
index=_internal source=*validation.log
```

Then, if you see results such as the entry below, the problem is with the App Certification server:
Click Stop at any time during the validation process if you think the process is not responding.

You can also switch to other tasks in Add-on Builder while validation is running. The validation process will continue to run in the background until the process has completed or you restart Splunk Enterprise.

**Download the add-on package**

Creating the installation package for an add-on includes packaging the props, transforms, tags, event types, lookups, modular inputs, setup screens, modular alerts, and prebuilt panels into a single new app and exporting it to a downloadable file.

If you have added or modified any files directly in the file system rather than using Add-on Builder, be sure you follow the recommendations in Modify files directly before you package your add-on.

splunkd must have OS-level access to all files that you manually copy to your add-on directory. splunkd permission is defined by the user running Splunk Enterprise. Insufficient access permission is indicated by a grayed-out Download Package button.

**To download your add-on package file**

1. On your add-on homepage, click **Validate & Package** on the Add-on Builder navigation bar.
2. Click **Download Package** to download the installation package for the add-on as an SPL file.
Use the Add-ons

Use the add-on

After you download the add-on package you created using Splunk Add-on Builder, you can use this add-on to get related data in.

The following tasks describe how to set up and configure different aspects of a data input in an add-on created by Splunk Add-on Builder. The example add-on requires an account and password for the data source.

Set up the add-on

The following setup tasks are performed in the Configuration page.

1. From the Splunk Web home page, click the name of the add-on in the left navigation panel.
2. Click Configuration.

Account settings

1. Create a account with a unique name in the Account field.
2. Enter the Username and Password of the account which you want to get data from.

Proxy settings

1. If you are using a proxy, check Enable and fill in the required fields. If you do not use a proxy, skip this section.
2. Select the type of proxy to use in the Proxy Type field.
3. Provide the proxy server address in the Host field.
4. Provide the proxy server port in the Port field. For example: 8081.
5. Provide a proxy username if you have one in the Username field.
6. If you provided a proxy username, type the proxy password in the Password field.
7. Check the Remote DNS resolution box if you want to perform DNS resolution through your proxy.

Logging Settings

1. To change the Log level, select a new level from the drop down menu.

Configure inputs

Configure the inputs to collect data via Splunk Web. Click Create New Input in the Inputs tab and configure the following.

1. Enter a Name, Interval and Index.
   - Name: The name of the input.
   - Index: The index in which you want the Splunk platform to store indexed data.
   - Interval: The number of seconds to wait before the Splunk platform runs the command again.
2. Choose the Checkpoint type and enter the checkpoint value. If you need more details on setting modular input checkpoints, see Data checkpoints in Developing Views and Apps for Splunk Web.
   - Checkpoint type: There are two checkpoint type, Auto and File.
     - If you choose Auto, the checkpoint file will be saved in the KVstore.
If you choose **File**, the checkpoint file will be saved under folder

$SPLUNK_HOME$/var/lib/splunk/modinputs

3. Restart the Splunk platform.
Advanced

Create a new data model

From version 2.2.0, Splunk add-on builder supports the user to map the data event to the data model you create. Users can design and maintain data models and use them in Splunk Add-on builder. Splunk recommends you to use Splunk web first and then modify the data model JSON file to follow the standard of Add-on builder.

To create data models which can be used by Splunk add-on builder, you need to understand

- What is data models and how to create a data model in Splunk platform.
- The format and semantics of their indexed data and are familiar with the Splunk search language. In building a typical data model, knowledge managers use knowledge object types such as lookups, transactions, search-time field extractions, and calculated fields.
- The data model standard of Splunk add-on builder. See syntax of data model for details.

Make sure you have sufficient access permissions to any files you place in your add-on directory.

Syntax of data model

After building data model using Splunk web, the generated JSON file cannot be used by Splunk Add-on builder directly, add the following fields to the existing JSON file.

**Required field**

objects.comment.tags

Syntax: $.objects[*].comment.tags

Description: It defines the tags of object in the data model. Eventtype which has the same tag(s) will be mapped to this data model.

**Optional fields**

objects.comment.description

Syntax: $.objects[*].comment.description

Description: The description of the data model.

object.fields.comment.description

Syntax: $.object[*].fields[*].comment.description:<string>

Description: The description of the data model field.

object.fields.comment.expected_values

Syntax: $.object[*].fields[*].comment.expected_values:<string>

Description: It defines the expected value(s) of the data model field. Splunk add-on builder verifies the expected value(s) when user validates the add-on.
Example: Create a data model named test

1. Create the data model using Splunk Web and name it as 'test'.
2. Open the test.json file under

$SPLUNK_HOME/etc/apps/<your_addon_folder>/default/data/models/test.json and add the field required by Splunk Add-on builder as follows

3. Save the file and then restart Splunk
4. The data model you create will be listed on the Select Data Models page. Follow the instruction on how to map to data model.

Modify files directly

For certain situations, you might want to modify your add-on outside of the Add-on Builder by modifying files directly. Follow these guidelines for best results:

- When modifying configuration files, for example to add stanzas, modify the version in the $SPLUNK_HOME/etc/apps/your_addon/default directory.
- If you need to add a configuration file, add the file to the $SPLUNK_HOME/etc/apps/your_addon/default directory.
- Make sure you have sufficient access permissions to any files you place in your add-on directory.
- Do not place files in the /local directory. Although the Add-on Builder creates copies of the configuration files for your add-ons in its /local directory, this is a working location and files can be overwritten. When packaging an add-on, the Add-on Builder merges any files from your add-on’s /default directory with its working copy.
Advanced modifications might affect the add-on’s eligibility for Splunk App Certification.

**Learn more**

For more information, see the following Splunk Enterprise documentation:

- About configuration files
- Configuration file precedence

**Python helper functions**

The Add-on Builder provides Python code templates for creating modular data inputs and alert actions, with helper functions for working with parameters.

Helper functions are Python 2 and Python 3 compatible.

You can only access input arguments using helper functions.

**Functions to get values from input parameters**

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>get_app_name()</td>
<td>Returns the name of the add-on.</td>
</tr>
<tr>
<td>get_input_type()</td>
<td>Returns the value of Input name on the Data Input Properties tab of the Inputs &amp; Parameters step.</td>
</tr>
<tr>
<td>get_input_stanza(input_stanza_name=None)</td>
<td>If input_stanza_name is provided, returns a dict containing all stanza parameters for the specified stanza name, or “None” if the stanza name does not exist. If input_stanza_name is not provided, returns all stanzas in a dict, with the stanza name as the key and detailed stanza parameters as the value. <strong>Note</strong> When input_stanza_name is not provided, one key-value pair is returned for multi-instance mode, and multiple key-value pairs are returned for single-instance mode.</td>
</tr>
<tr>
<td>get_input_stanza_names()</td>
<td>Returns all stanza names for this input. A single string value is returned for multi-instance mode. A list of stanza names is returned for single-instance mode.</td>
</tr>
<tr>
<td>get_arg(arg_name, input_stanza_name=None)</td>
<td>Returns an argument value based on arg_name and input_stanza_name. Arguments are defined on the Data Input Parameters tab of the Inputs &amp; Parameters step. arg_name refers to the input field’s internal name. For multi-instance mode, input_stanza_name is not necessary, and the returned arg value is “None”. For single-instance mode, if input_stanza_name is ”None”, a dict with the stanza name as the key and the arg value as the value is returned. If input_stanza_name is not ”None”, a single value of ”None” is returned.</td>
</tr>
<tr>
<td>get_sourcetype(input_stanza_name=None)</td>
<td>Returns the source type.</td>
</tr>
<tr>
<td>get_output_index(input_stanza_name=None)</td>
<td>Returns the output index.</td>
</tr>
</tbody>
</table>
## Functions to get values from setup parameters

Helper functions to get credential accounts are used in rare cases. If you need an account in your modular input code, add a Global Account field to your setup parameters, then use `helper.get_arg("internal_name")`, where `internal_name` is the internal name of the Global Account field, to retrieve the value.

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>get_proxy()</code></td>
<td>If Proxy settings is selected, returns a dict: <code>{ &quot;proxy_url&quot;: ..., &quot;proxy_port&quot;: ..., &quot;proxy_username&quot;: ..., &quot;proxy_password&quot;: ..., &quot;proxy_type&quot;: ..., &quot;proxy_rdns&quot;: ... }</code> Otherwise, an empty dict is returned.</td>
</tr>
<tr>
<td><code>get_user_credential_by_username(username)</code></td>
<td>If <code>username</code> exists, returns the dict: <code>{ &quot;name&quot;: account_id, &quot;username&quot;: username, &quot;password&quot;: password }</code> Otherwise, returns &quot;None&quot;.</td>
</tr>
<tr>
<td><code>get_user_credential_by_id(account_id)</code></td>
<td>If <code>account_id</code> exists, returns the dict: <code>{ &quot;name&quot;: account_id, &quot;username&quot;: username, &quot;password&quot;: password }</code> Otherwise, returns &quot;None&quot;.</td>
</tr>
<tr>
<td><code>get_global_setting(var_name)</code></td>
<td>Returns a global setting value for <code>var_name</code>. This method is used to get custom settings.</td>
</tr>
<tr>
<td><code>get_log_level()</code></td>
<td>Returns the log level that was selected. If not set, returns &quot;INFO&quot;.</td>
</tr>
<tr>
<td><code>set_log_level(level)</code></td>
<td>Sets the log level to <code>level</code>.</td>
</tr>
<tr>
<td><code>log(msg)</code></td>
<td>Logs a message using the configured logging level.</td>
</tr>
<tr>
<td><code>log_debug(msg)</code></td>
<td>Logs a message using the <code>logging.DEBUG</code> level.</td>
</tr>
<tr>
<td><code>log_info(msg)</code></td>
<td>Logs a message using the <code>logging.INFO</code> level.</td>
</tr>
<tr>
<td><code>log_warning(msg)</code></td>
<td>Logs a message using the <code>logging.WARNING</code> level.</td>
</tr>
<tr>
<td><code>log_error(msg)</code></td>
<td>Logs a message using the <code>logging.ERROR</code> level.</td>
</tr>
<tr>
<td><code>log_critical(msg)</code></td>
<td>Logs a message using the <code>logging.CRITICAL</code> level.</td>
</tr>
</tbody>
</table>

## Functions for checkpoints

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>save_check_point(key, state)</code></td>
<td>Saves the checkpoint <code>{key: state}</code> to the Splunk KV store.</td>
</tr>
<tr>
<td><code>delete_check_point(key)</code></td>
<td>Deletes the checkpoint corresponding to <code>key</code>.</td>
</tr>
<tr>
<td><code>get_check_point(key)</code></td>
<td>Get the value of the checkpoint corresponding to <code>key</code>.</td>
</tr>
<tr>
<td><code>batch_save_check_point(states)</code></td>
<td>Saves a batch of <code>{key: state}</code> pairs, where states is a dict, the <code>key</code> is the checkpoint state key, and <code>state</code> is the value.</td>
</tr>
</tbody>
</table>
Functions for alert actions

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
</table>
| `addevent(event_message, sourcetype)` | Adds a custom event to the Adaptive Response Framework.  
  - `event_message`: The event message.  
  - `sourcetype`: The source type of this event. |
| `writeevents(index, host, source)`   | Flushes the custom Adaptive Response Framework events. Call this function only once in your adaptive response.  
  - `index`: The index that stores these events.  
  - `host`: The host fields of these events.  
  - `source`: The source fields of these events. |
| `get_events()`   | Gets an event list that contains all of the search results when this alert is triggered.                                                     |
| `get_param(param_name)` | Gets the parameter value when the alert is triggered for `param_name`, which is the internal name of the alert parameter.  |

Other functions

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
</table>
| `send_http_request(url, method, parameters=None, payload=None, headers=None, cookies=None, verify=True, cert=None, timeout=None, use_proxy=True)` | Sends an HTTP request and returns the response.  
  - `url`: URL for the new Request object.  
  - `method`: Method for the new Request object (`GET` | `POST` | `PUT` | `DELETE`).  
  - `parameters`: Optional. Dictionary or bytes to be sent in the query string for the request.  
  - `payload`: Optional. Dictionary, bytes, or file-like object to send in the body of the request.  
  - `headers`: Optional. Dictionary of HTTP headers to send with the request.  
  - `cookies`: Optional. Dict or CookieJar object to send with the request.  
  - `verify`: Optional. Indicates whether the SSL certificate will be verified. A CA_BUNDLE path can also be provided.  
  - `cert`: Optional. If a string, contains the path to the SSL client cert (.pem) file. If Tuple, a (`'cert'`, `'key'`) pair.  
  - `timeout` (float or tuple): Optional. Specifies how long to wait for the server to send data, as a float or a (connect timeout, read timeout) tuple.  
  - `use_proxy`: Optional. Indicates whether to use a proxy. If True, the proxy in the Add-on Builder Configuration settings is used. |
| `new_event(data, time=None, host=None, index=None, source=None, sourcetype=None, done=True, unbroken=True)` | Creates a new event. This function is used to index data in Splunk Enterprise.  
  The example in the code template shows `ew.write_event(event)`. Refer to splunklib.modularinput.Event in splunklib. |
Release Notes

Release notes

Version 7.2.0 was released on May 20, 2020.

What's new in version 3.0.2

- Added a python.version flag to all .conf files.

Known Issues

Version 3.0.2 of the Splunk Add-on Builder contains the following known issues. If no issues appear, no issues have been reported:

Fixed Issues

Version 3.0.2 of Splunk Add-on Builder fixes the following issues. If no issues appear, no issues have been reported:

Third-party software credits

Some of the components included in Splunk Add-on Builder are licensed under free or open source licenses. We wish to thank the contributors to these projects.

DefiantJS

https://github.com/hbi99/defiant.js

DefiantJS version 1.4.0

Copyright (c) 2013 Hakan Bilgin

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

httplib2

https://github.com/httplib2

httplib2 version 0.9.2

Copyright (c) 2006 by Joe Gregorio

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to
use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to
whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE,
ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

Immutable

https://github.com/facebook/immutable-js

Immutable version 3.8.1

MIT License

Copyright (c) 2014-present, Facebook, Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to
use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to
whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE,
ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.
**JavaScript Cookie**

https://github.com/js-cookie/js-cookie

JavaScript Cookie version 2.1.2

Copyright 2014 Klaus Hartl

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

**Jinja2**

http://jinja.pocoo.org/

Jinja2 version 2.8

Copyright (c) 2009 by the Jinja Team, see AUTHORS for more details.

Some rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- The names of the contributors may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

jsl

https://github.com/aromanovich/jsl

jsl version 0.2.4

Copyright (c) 2016 by Anton Romanovich.

Some rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

- The names of the contributors may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

JSONPath RW

https://github.com/kennknowles/python-jsonpath-rw

JSONPath RW version 1.4.0

Copyright 2013- Kenneth Knowles

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the
specific language governing permissions and limitations under the License.

jsonschema

https://github.com/Julian/jsonschema

jsonschema version 2.6.0

Copyright (c) 2013 Julian Berman

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Lodash

https://github.com/lodash/lodash

Lodash versions 4.17.3

The MIT License

Copyright JS Foundation and other contributors <https://js.foundation/>

Based on Underscore.js, copyright Jeremy Ashkenas, DocumentCloud and Investigative Reporters & Editors <http://underscorejs.org/>

This software consists of voluntary contributions made by many individuals. For exact contribution history, see the revision history available at https://github.com/lodash/lodash

The following license applies to all parts of this software except as documented below:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
SOFTWARE.

PySocks

https://github.com/Anorov/PySocks

PySocks version 1.6.5

Copyright 2006 Dan-Haim. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. 3. Neither the name of Dan Haim nor the names of his contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY DAN HAIM "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL DAN HAIM OR HIS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

React

https://github.com/facebook/react

React version 15.5.4

Copyright (c) Facebook, Inc. and its affiliates.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
React Redux

https://github.com/reactjs/react-redux

React Redux version 4.4.8

The MIT License (MIT)

Copyright (c) 2015-present Dan Abramov

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Redux

https://github.com/reactjs/redux

Redux version 3.6.0

Copyright (c) 2015-present Dan Abramov

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
redux-actions

https://github.com/acdlite/redux-actions

redux-actions version 1.2.0

Copyright (c) 2016 Andrew Clark

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

redux-immutable

https://github.com/gajus/redux-immutable

redux-immutable version 3.0.9

Copyright (c) 2016, Gajus Kuizinas (http://gajus.com/) All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

• Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
• Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
• Neither the name of the Gajus Kuizinas (http://gajus.com/) nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL ANY PARTY BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
redux-observable

https://github.com/redux-observable/redux-observable

redux-observable version 0.12.2

Copyright (c) 2016 Ben Lesh, Jay Phelps, and redux-observable contributors.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Regulex

https://github.com/JexCheng/regulex

Regulex version 0.0.2

Copyright (c) 2014 Jex

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Requests

https://github.com/kennethreitz/requests

Requests version 2.22.0
COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Select2

http://select2.github.io/select2/

Select2 version 3.4.6

Copyright 2012 Igor Vaynberg

This software is licensed under the Apache License, Version 2.0 (the "Apache License") or the GNU General Public License version 2 (the "GPL License"). You may choose either license to govern your use of this software only upon the condition that you accept all of the terms of either the Apache License or the GPL License.

You may obtain a copy of the Apache License and the GPL License in the LICENSE file, or at:


Unless required by applicable law or agreed to in writing, software distributed under the Apache License or the GPL License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the Apache License and the GPL License for the specific language governing permissions and limitations under the Apache License and the GPL License.

simpleyaml

https://pypi.python.org/pypi/simpleyaml

simpleyaml version 1.0

Copyright (c) 2012 Kirill Simonov and Reuven V. Gonzales

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
Sortable

https://github.com/RubaXa/Sortable

Sortable version 1.5.1

Copyright (c) 2019 All contributors to Sortable https://github.com/SortableJS

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

SortedContainers

https://github.com/grantjenks/sorted_containers/

SortedContainers version 1.5.2

Copyright 2014-2016 Grant Jenks

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

uuid

https://github.com/kelektiv/node-uuid

uuid version 3.0.1

Copyright (c) 2010-2016 Robert Kieffer and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to
whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The UCC 3.0.2 shared library

The UCC 3.0.2 shared library contains the following 3rd party software:

**jquery**

Version 2.1.0

Copyright 2014 jQuery Foundation and other contributors

http://jquery.com/

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

**select2**

Version 3.4.6 MIT license

Copyright 2012 Igor Vaynberg

This software is licensed under the Apache License, Version 2.0 (the "Apache License") or the GNU General Public License version 2 (the "GPL License"). You may choose either license to govern your use of this software only upon the condition that you accept all of the terms of either the Apache License or the GPL License.

You may obtain a copy of the Apache License and the GPL License in the LICENSE file, or at:
Unless required by applicable law or agreed to in writing, software distributed under the Apache License or the GPL License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the Apache License and the GPL License for the specific language governing permissions and limitations under the Apache License and the GPL License.

**jquery-ui**

Version 1.10.4

Copyright 2014 jQuery Foundation and other contributors, http://jqueryui.com/

This software consists of voluntary contributions made by many individuals (AUTHORS.txt, http://jqueryui.com/about) For exact contribution history, see the revision history and logs, available at http://jquery-ui.googlecode.com/svn/

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

**moment.js**

Version 2.8.3 Copyright (c) 2011-2015 Tim Wood, Iskren Chernev, Moment.js contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
jqTree

Version 0.19

Apache License

Version 2.0, January 2004

http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.
2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

You must give any other recipients of the Work or Derivative Works a copy of this License; and You must cause any modified files to carry prominent notices stating that You changed the files; and You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works; and If the Work is a modification of a Work, or is derived from the Work or a work containing a notice within the meaning of this License, You must do one of the following: provide a complete copy of the经理 Notice file; include a notice in the Source form of your modifications stating that You have no additional terms; deletes the NOTICE file from all distributions; or,Altering the contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License. 5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages
of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

bootstrap

Version 2.3.1

The MIT License (MIT)

Copyright (c) 2011-2014 Twitter, Inc

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

underscore

Version 1.6.0 Copyright (c) 2009-2014 Jeremy Ashkenas, DocumentCloud and Investigative Reporters & Editors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
Backbone.Validation

Version 0.9.1 Copyright © 2014 Thomas Pedersen, http://thedersen.com

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to
use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to
whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE,
ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

jquery-resize-plugin

Version 1.1

Copyright (c) 2010 "Cowboy" Ben Alman

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to
use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to
whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE,
ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

low-pro-for-jquery

Version 1.0.0

Copyright (c) 2009 Dan Webb

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to
use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to
whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

`sax.js`

Version 1.1.0

The ISC License

Copyright (c) Isaac Z. Schlueter and Contributors

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

== `String.fromCharCode` by Mathias Bynens used according to terms of MIT License, as follows:

Copyright Mathias Bynens <https://mathiasbynens.be/>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

`lodash`

Version 4.16.4

https://github.com/lodash/lodash
The MIT License

Copyright JS Foundation and other contributors <https://js.foundation/>

Based on Underscore.js, copyright Jeremy Ashkenas, DocumentCloud and Investigative Reporters & Editors <http://underscorejs.org/>

This software consists of voluntary contributions made by many individuals. For exact contribution history, see the revision history available at https://github.com/lodash/lodash

The following license applies to all parts of this software except as documented below:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

highcharts

Version 4.0.4

Commercial license

https://github.com/highcharts/highcharts/blob/master/license.txt:

"This is the repository for Highsoft’s JavaScript charting libraries: Highcharts, Highstock and Highmaps.

The libraries are available under different licenses depending on whether it is intended for commercial/government use, or for a personal or non-profit project.

Read more on licensing alternatives here:

- Highcharts: https://shop.highsoft.com/highcharts
- Highstock: https://shop.highsoft.com/highstock
- Highmaps: https://shop.highsoft.com/highmaps"

intro.js

Version 1.0.0 Copyright (C) 2012-2017 Afshin Mehrabani

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed. See the full GNU license.
Cloud Connect Engine 2.0.2 shared library

This Cloud Connect Engine 2.02 shared library contains the following 3rd party software:

**jsonschema**

https://github.com/Julian/jsonschema

jsonschema version 2.6.0

Copyright (c) 2013 Julian Berman

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

**jinja2**

http://jinja.pocoo.org/

Version 2.10.1

https://github.com/pallets/jinja/blob/master/LICENSE

Copyright 2007 Pallets

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

**jsonp RW**

https://github.com/kennknowles/python-jsonpath-rw

JSONPath RW version 1.4.0

Copyright 2013- Kenneth Knowles

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

**httplib2**

https://github.com/jcgregorio/httplib2 Version 0.14.0

MIT license

Copyright (c) 2006 by Joe Gregorio

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

**splunk-sdk**

http://github.com/splunk/splunk-sdk-python

Version 1.6.6
1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

sortedcontainers

http://www.grantjenks.com/docs/sortedcontainers/

Version 2.1.0

Apache License

Version 2.0, January 2004

http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent
(50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   You must give any other recipients of the Work or Derivative Works a copy of this License; and
   You must cause any modified files to carry prominent notices stating that You changed the files; and
   You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
   If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You
distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

munch

https://github.com/Infinidat/munch

munch version 2.3.2

Copyright (c) 2010 David Schoonover
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to
use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to
whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE,
ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

functools32

https://pypi.python.org/pypi/functools32/3.2.3-2 Version 3.2.3-2

Python License (Python-2.0) Python License, Version 2 (Python-2.0) PYTHON SOFTWARE FOUNDATION LICENSE
VERSION 2 --------------------------------------------- 1. This LICENSE AGREEMENT is between the Python Software
Foundation ("PSF"), and the Individual or Organization ("Licensee") accessing and otherwise using this software
("Python") in source or binary form and its associated documentation. 2. Subject to the terms and conditions of this
License Agreement, PSF hereby grants Licensee a nonexclusive, royalty-free, world-wide license to reproduce, analyze,
test, perform and/or display publicly, prepare derivative works, distribute, and otherwise use Python alone or in any
derivative version, provided, however, that PSF's License Agreement and PSF's notice of copyright, i.e., "Copyright (c)
any derivative version prepared by Licensee. 3. In the event Licensee prepares a derivative work that is based on or
incorporates Python or any part thereof, and wants to make the derivative work available to others as provided herein,
then Licensee hereby agrees to include in any such work a brief summary of the changes made to Python. 4. PSF is
making Python available to Licensee on an "AS IS" basis. PSF MAKES NO REPRESENTATIONS OR WARRANTIES,
EXPRESS OR IMPLIED. BY WAY OF EXAMPLE, BUT NOT LIMITATION, PSF MAKES NO AND DISCLAIMS ANY
REPRESENTATION OR WARRANTY OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR
THAT THE USE OF PYTHON WILL NOT INFRINGE ANY THIRD PARTY RIGHTS. 5. PSF SHALL NOT BE LIABLE TO
LICENSEE OR ANY OTHER USERS OF PYTHON FOR ANY INCIDENTAL, SPECIAL, OR CONSEQUENTIAL
DAMAGES OR LOSS AS A RESULT OF MODIFYING, DISTRIBUTING, OR OTHERWISE USING PYTHON, OR ANY
DERIVATIVE THEREOF, EVEN IF ADVISED OF THE POSSIBILITY THEREOF. 6. This License Agreement will
automatically terminate upon a material breach of its terms and conditions. 7. Nothing in this License Agreement shall be
deemed to create any relationship of agency, partnership, or joint venture between PSF and Licensee. This License
Agreement does not grant permission to use PSF trademarks or trade name in a trademark sense to endorse or promote
products or services of Licensee, or any third party. 8. By copying, installing or otherwise using Python, Licensee agrees
to be bound by the terms and conditions of this License Agreement. BEOPEN.COM LICENSE AGREEMENT FOR PYTHON 2.0

------------------------------------------- BEOPEN PYTHON OPEN SOURCE LICENSE AGREEMENT VERSION 1 1. This
LICENSE AGREEMENT is between BeOpen.com ("BeOpen"), having an office at 160 Saratoga Avenue, Santa Clara, CA
95051, and the Individual or Organization ("Licensee") accessing and otherwise using this software in source
or binary form and its associated documentation ("the Software"). 2. Subject to the terms and conditions of this
BeOpen Python License Agreement, BeOpen hereby grants Licensee a non-exclusive, royalty-free,
world-wide license to reproduce, analyze, test, perform and/or display publicly, prepare derivative works,
distribute, and otherwise use the Software alone or in any derivative version, provided, however, that the
BeOpen Python License is retained in the Software, alone or in any derivative version prepared by Licensee.
3. BeOpen is making the Software available to Licensee on an "AS IS" basis. BEOPEN MAKES NO REPRESENTATIONS
IMPORTANT: PLEASE READ THE FOLLOWING AGREEMENT CAREFULLY. BY CLICKING ON "ACCEPT" WHERE INDICATED BELOW, OR BY COPYING, INSTALLING OR OTHERWISE USING PYTHON 1.6, beta 1 SOFTWARE, YOU ARE DEEMED TO HAVE AGREED TO THE TERMS AND CONDITIONS OF THIS LICENSE AGREEMENT. 1. This LICENSE AGREEMENT is between the Corporation for National Research Initiatives, having an office at 1895 Preston White Drive, Reston, VA 20191 ("CNRI"), and the Individual or Organization ("Licensee") accessing and otherwise using Python 1.6, beta 1 software in source or binary form and its associated documentation, as released at the www.python.org Internet site on August 4, 2000 ("Python 1.6b1"). 2. Subject to the terms and conditions of this License Agreement, CNRI hereby grants Licensee a non-exclusive, royalty-free, world-wide license to reproduce, analyze, test, perform and/or display publicly, prepare derivative works, distribute, and otherwise use Python 1.6b1 alone or in any derivative version, provided, however, that CNRIs License Agreement is retained in Python 1.6b1, alone or in any derivative version prepared by Licensee. Alternately, in lieu of CNRIs License Agreement, Licensee may substitute the following text (omitting the quotes): "Python 1.6, beta 1, is made available subject to the terms and conditions in CNRIs License Agreement. This Agreement may be located on the Internet using the following unique, persistent identifier (known as a handle): 1895.22/1011. This Agreement may also be obtained from a proxy server on the Internet using the URL:http://hdl.handle.net/1895.22/1011". 3. In the event Licensee prepares a derivative work that is based on or incorporates Python 1.6b1 or any part thereof, and wants to make the derivative work available to the public as provided herein, then Licensee hereby agrees to indicate in any such work the nature of the modifications made to Python 1.6b1. 4. CNRI is making Python 1.6b1 available to Licensee on an "AS IS" basis. CNRI MAKES NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED. BY WAY OF EXAMPLE, BUT NOT LIMITATION, CNRI MAKES NO AND DISCLAIMS ANY REPRESENTATION OR WARRANTY OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF PYTHON 1.6b1 WILL NOT INFRINGE ANY THIRD PARTY RIGHTS. 5. CNRI SHALL NOT BE LIABLE TO LICENSEE OR ANY OTHER USERS OF THE SOFTWARE FOR ANY INCIDENTAL, SPECIAL, OR CONSEQUENTIAL DAMAGES OR LOSS AS A RESULT OF USING, MODIFYING OR DISTRIBUTING PYTHON 1.6b1, OR ANY DERIVATIVE THEREOF, EVEN IF ADVISED OF THE POSSIBILITY THEREOF. 6. This License Agreement will automatically terminate upon a material breach of its terms and conditions. 7. This License Agreement shall be governed by and interpreted in all respects by the law of the State of Virginia, excluding conflict of law provisions. Nothing in this License Agreement shall be deemed to create any relationship of agency, partnership, or joint venture between CNRI and Licensee. This License Agreement does not grant permission to use CNRI trademarks or trade name in a trademark sense to endorse or promote products or services of Licensee, or any third party. 8. By clicking on the "ACCEPT" button where indicated, or by copying, installing or otherwise using Python 1.6b1, Licensee agrees to be bound by the terms and conditions of this License Agreement. ACCEPT CWI LICENSE AGREEMENT FOR PYTHON 0.9.0 THROUGH 1.2

Copyright (c) 1991 - 1995, Stichting Mathematisch Centrum Amsterdam, The Netherlands. All rights reserved. Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of Stichting Mathematisch Centrum or CWI not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. STICHTING
futures

https://pypi.python.org/pypi/futures

Version 3.0.5

The 2-Clause BSD License

SPDX short identifier: BSD-2-Clause

OSI Approved License Logo

Note: This license has also been called the "Simplified BSD License" and the "FreeBSD License". See also the 3-clause BSD License.

Copyright <YEAR> <COPYRIGHT HOLDER>

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

configparser

https://pypi.org/project/configparser/3.7.4/

Version 3.7.4

Copyright Jason R. Coombs
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to
use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to
whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE,
ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

future

https://pypi.org/project/future/

Version 0.17.1

Copyright (c) 2013-2019 Python Charmers Pty Ltd, Australia

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to
use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to
whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE,
ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

six

https://pypi.org/project/six/

Version 1.12.0

Copyright (c) 2010-2019 Benjamin Peterson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to
use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to
whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

lib-solutions-python shared library version 2.0.1

The lib-solutions-python shared library contains the following 3rd party software:

requests

https://pypi.python.org/pypi/requests

Version 2.22.0

Copyright 2019 Kenneth Reitz

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

sortedcontainers

https://pypi.python.org/pypi/sortedcontainers

Version 2.1.0

Copyright 2014-2016 Grant Jenks ttp://www.grantjenks.com/docs/sortedcontainers/

Version 2.1.0

Apache License

Version 2.0, January 2004

http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.
"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- You must give any other recipients of the Work or Derivative Works a copy of this License; and
- You must cause any modified files to carry prominent notices stating that You changed the files; and
- You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.
- You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution." "Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of Your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

PyYaml

https://pypi.org/project/PyYAML/

Version 5.1.1, 5.2 Copyright 2016 Reuven V. Gonzales

MIT http://opensource.org/licenses/MIT

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

schematics

https://pypi.python.org/pypi/schematics 2.1.0 BSD https://github.com/schematics/schematics/blob/development/LICENSE "Copyright (c) 2013?2016 Schematics Authors; see AUTHORS for details All rights reserved." and BSD license. (Legal): Removed the following copyright attribution notice, as it: 1. is not what is listed with this component; and 2. does not fully capture all of the contributors associated with this component. Copyright 2016 James Dennis, Jökull Sólberg, Jóhann Þorvaldur Bergþórsson, Kalle Tuure

Note: UCC also relies on lib-solutions-pythons 1.0.15, refer to its license as well