Splunk® Security Content
How to Use Splunk Security Content 3.52.0

Generated: 11/09/2022 10:31 am

Table of Contents

About Splunk Security Content

About Splunk Security Content

Splunk Security Content delivers security analysis guides called Analytic Stories. These Analytic Stories, which are authored by the Splunk Security Research Team, give you advice on how to use Splunk Enterprise Security (ES) to investigate and take action on new threats that Splunk ES detects in your environment. They contain the searches you need to implement the story in your own Splunk ES environment. Each story also provides an explanation of what the search achieves and how to convert a search into adaptive-response actions, where appropriate.

Download Splunk Security Content from Splunkbase.

Use Splunk Security Content

How to use Splunk Security Content

From the Splunk Security Content menu bar, you can navigate to the following pages:

- Content Library to view the dashboard of Analytic Stories and search summaries
- Feedback Center to send feedback directly to the Splunk Security Research Team
- Usage Details to see what your users are doing inside your instance of ESCU

If you use Splunk Enterprise Security or Splunk Security Essentials, you can access Analytic Stories through those apps.

- See Manage Analytic Stories through the use case library in Splunk Enterprise Security in the Splunk Enterprise Security *Administer Splunk Enterprise Security* manual to enable correlation searches.
- See About the Splunk Security Essentials app in the Splunk Security Essentials *Use Splunk Security Essentials* manual.

Content Library

Access the **Analytic Stories Stats** tab to explore the Analytic Stories included with Splunk Security Updates using the Category, CIS Critical Security Control, or Kill Chain Phase mapping.

Access the **Search Summary** tab to see the searches associated with an Analytic Story and explore them based on their CIS Critical Security Control mapping or by search type.

Analytic Story Detail

The **Analytic Story Detail** view provides information on how to use Splunk ES to address a particular threat.

Splunk Security Content categorizes the stories according to the table below.

Category	Analytic Stories
Malware	Brand Monitoring Data Protection Host Redirection Lateral Movement Malicious PowerShell Ransomware SQL Injection Suspicious DNS Traffic Suspicious Emails Suspicious WMI Use Unusual Processes Windows Log Manipulation Windows Persistence Techniques
Known vulnerabilities	Apache Struts Vulnerability DNS Amplification Attacks JBOSS Vulnerability Monitor for Updates Splunk Enterprise Vulnerability

Category	Analytic Stories
Best practices	Account Monitoring and Controls Asset Tracking Router & Infrastructure Security Monitor Backup Solution Monitor for Unauthorized Software Use of Cleartext Protocols Prohibited Traffic Allowed or Protocol Mismatch

Analytic Stories provide you with tactics, techniques, and methodologies to assist with detection, investigation, and response. They include easy-to-read background information, key context for motivations and risks associated with the attack techniques in question, and pragmatic advice on how to combat those techniques.

Each story is mapped to various frameworks, including MITRE ATT&CK, Lockheed Martin Kill Chain phases, CIS controls, and NIST, and include the following content objects:

1. **Detection:** OOTB detection techniques in the form of detection searches or machine-learning models
2. **Investigation:** Searches and/or Splunk Phantom playbooks that help the analyst determine whether a notable event is true-positive. For example, the analyst may wish to review additional notables related to the participating entity (additional detections). They may also need to gather collaborative evidence and additional contextual information.
3. **Response:** These help the analyst conduct specific response actions to remediate the incident.

Analytic Stories are categorized by use case and can be accessed via the Splunk ES Use Case Library or Splunk Security Content.

Select any search to view its search name, description, kill chain phase, and details.

Customize to your Environment

Release 1.0.46 introduced `input(pre-filter)` and `output(post-filter)` macros for each of our detection searches. These macros let you update a macro definition once and then apply the new definition across all detections that leverage that macro. These changes will be local to your Splunk environment.

- **input(pre-filter):** This macro specifies your environment-specific configurations (index, source, sourcetype, etc.) to get the specific data sources that you require. Replace the macro definition with configurations for your Splunk environment.

- **output(post-filter):** This macro specifies your environment-specific values (dest, user, etc,), to filter out known false positives. Replace the macro definition with values that you'd like to exclude from detection results.

> Coming soon is an improved naming convention that will be consistent across all of our detections, investigations, and baselines.

Feedback Center

Access the **Feedback Center** to send feedback directly to the Splunk Security Research Team. Contact us at research@splunk.com to send us support requests, bug reports, or questions directly to the Splunk Security Research Team. Please specify your request type and/or the title of any related Analytic Stories.

Usage Details

Access **Usage Details** to see how your team is using Splunk Security Content. You can access details such as the following:

- The searches your team runs most frequently
- The types of searches your team runs, including the names of the searches and the average and total run times

Use the Splunk Machine Learning Toolkit (MLTK) with Splunk Security Content

Install and set up the Splunk Machine Learning Toolkit

The Splunk Machine Learning Toolkit (MLTK) enables users to create, validate, manage, and operationalize machine-learning models through a guided user interface. Many of the searches provided in Splunk Security Content use MLTK to create models and enhance performance.

The current version of the Splunk Machine Learning Toolkit is 4.2.0 and requires Splunk Enterprise 6.6 or later or Splunk Cloud and Python for Scientific Computing add-on version 1.3 or 1.4.

To get started, download MLTK from Splunkbase and then visit this page for installation instructions.

Configure Splunk Enterprise Security to use the Machine Learning Toolkit

You can configure Splunk Enterprise Security (ES) to use the Machine Learning Toolkit (MLTK). MLTK enables users to create, validate, manage, and operationalize machine learning models through a guided user interface. See About the Machine Learning Toolkit in the Splunk Machine Learning Toolkit *User Guide*.

Using a version of ES that is 6.0.0 or higher

If you are using ES 6.0.0 or higher, MLTK is included in the installer. There are no additional steps. See Release Notes for Splunk Enterprise Security.

Using a version of ES that is lower than 6.0.0

If you are using a version of ES that is lower than 6.0.0, complete the following steps to configure ES to use MLTK.

After downloading MLTK from Splunkbase, visit this page for installation instructions, then follow the steps below to import MLTK for use with ES.

1. On the Enterprise Security toolbar, browse to Configure > General > App Imports Update

2. Edit the `update_es input`

3. In the field for "Application Regular Expression," add the following to the end of the existing string: |(`Splunk_ML_Toolkit`)

4. Click "Save"

Detailed documentation on importing an app/add-on can be found at https://docs.splunk.com/Documentation/ES/5.2.2/Install/ImportCustomApps#Import_add-ons_with_a_different_naming_convention

Use Splunk SOAR playbooks and workbooks from the Risk Notable Playbook Pack

Get started with the Risk Notable Playbook Pack for Splunk SOAR

This collection of playbooks and workbooks guides analysts through investigations of risk notables within Splunk SOAR. Risk notables are aggregates of risk anomalies within Splunk Enterprise Security. See Analyze risk in Splunk Enterprise Security in the *Use Splunk Enterprise Security* manual. As an analyst, learn how to use the workbooks, understand the playbooks, and explore customizing the playbooks.

> The playbook pack must be used with the latest release of Splunk Security Content.

Check prerequisites for the playbook pack

Before you use the playbook pack, verify that you have these dependencies:

- Splunk SOAR (Cloud) or (On-premises)
- Splunk Enterprise Security with assets and identities. See Manage assets and identities in Splunk Enterprise Security in the *Administer Splunk Enterprise Security* manual.
- Splunk Enterprise Security with the risk analysis frame working producing risk notables. See Analyze risk in Splunk Enterprise Security in the *User Splunk Enterprise Security* manual.
- Notables you produce from Splunk Enterprise Security must include these fields:
  - `risk_object`
  - `event_id`
  - `info_min_time`
  - `info_max_time`
- Use one of these apps from Splunkbase to bring Splunk Enterprise Security notable events into Splunk SOAR (Cloud) or (On-premises):
  - Splunk App for SOAR Export. Configure the multi-value field settings of Splunk App for SOAR Export to consolidate events into a single artifact. See About the Splunk App for SOAR Export and Configure how Splunk Phantom and Splunk SOAR handle multivalve fields in Splunk ES notable events in the *Use the Splunk App for SOAR Export to Forward Events* manual.
  - Splunk App for SOAR. Use this query in the *on poll* settings to find notable events in the correct fields:

```
`notable`
| search eventtype=risk_notables
| fields _time, event_hash, event_id, host, info_min_time, info_max_time, risk_object, risk_object_type,
risk_score, rule_description, rule_id, rule_name, search_name, source, splunk_server, urgency
```

- Splunk Enterprise Security with assets and identities (optional). See Manage assets and identities in Splunk Enterprise Security in the *Administer Splunk Enterprise Security* manual. The `splunk_enterprise_security_tag_assets_and_identities` playbook relies on this framework, and the `risk_notable_auto_containment` playbook uses resulting tags.

Deploy the playbook pack

Verify these deployment steps are done before you use the playbook pack:

- Because the playbook pack follows a five-point scale of severity based on Splunk Enterprise Security, a Splunk SOAR admin must add the severity levels "Critical" and "Informational" to the default severities of "High," "Medium," and "Low." See Create custom severity names in the *Administer Splunk SOAR (Cloud)* manual.
- Because the playbook pack uses the `risk_notable` label based on event types with the same names within Splunk Enterprise Security, a Splunk SOAR admin must add the `risk_notable` label. See Create a label in the *Administer Splunk SOAR (Cloud)* manual.
- Configure the base URL for Splunk SOAR.
- (Recommended step.) Copy all playbooks to a repository other than `community`, like `local`. See Configure a source control repository for your Splunk SOAR (Cloud) playbooks in the *Administer Splunk SOAR (Cloud)* manual. Update the matching sub-playbook calls to reference the correct repository, as well as the references in workbooks.
- If your Splunk asset on SOAR is not called **splunk**, change the asset name in the playbook to match the name of your Splunk asset.
- Splunk Web is configured on a port other than 443, like 8000, then includes the specified port directly after the hostname in these items:
    - The block "format es url" in the `risk_notable_preprocess` playbook
    - The block "format summary note" in the `risk_notable_import_data` playbook

Find playbooks in Splunk SOAR

To locate the playbooks from the playbook pack in Splunk SOAR (Cloud) or (On-premises), follow these steps:

1. From the Splunk SOAR (Cloud) or (On-premises) menu, select **Playbooks**.
2. Select **Update from Source Control** > **community** > **Update**.
3. Filter the **Category** column to **Risk Notable** to see all core playbooks.
4. Filter the **Tags** column to **risk_notable** to see all utility playbooks.
5. (Recommended step.) Copy the playbooks to the **local** repository so you can customize them.

Workbooks in the pack

Workbooks are guided analyst workflows with phases and tasks that can recommend actions and playbooks. This pack includes three workbooks.

Workbook	Description	Phase	Tasks	Workbook playbooks	Suggested playbooks

Risk Investigation	Guide the analyst from taking ownership of an investigation through rendering a verdict and selecting a response plan.	Initial Triage	Preprocess
			Investigate
			Render
			Verdict
				`risk_notable_investigate`	`risk_notable_preprocess`
					`risk_notable_import_data`
					`start_investigation`
					`risk_notable_enrich`
					`risk_notable_merge_events`
					`risk_notable_verdict`

Risk Response	Follow tasks to review suspect indicators, then select assets and users that need protection.	Mitigate	Block Indicators
			Protect Assets and Users
				`risk_notable_mitigate`	`risk_notable_review_indicators`
					`risk_notable_block_indicators`
					`risk_notable_protect_assets_and_users`

Workbook	Description	Phase	Tasks	Workbook playbooks	Suggested playbooks

Risk Recovery	Respond to confirmed incidences by documenting clean-up steps and closing out investigations.	Restore operations	Eradicate threats
			Undo containments
			Close investigations
				N/A	`risk_notable_auto_undo_containment`
					`reset_entity_risk`
					`splunk_enterprise_security_close_investigat`

See descriptions of playbooks in the Risk Notable Playbook Pack

The descriptions of playbooks included in this playbook pack are in this table:

Name	Description	Additional information
`risk_investigate`	This playbook checks for the Risk Investigation workbook, updates tasks, and takes notes.	Set this playbook to run in Active mode on the **Risk Notable** label in Splunk SOAR. To configure this playbook to automatically add notes, see the Playbook outputs section of Use the risk notable playbook pack to investigate a risk notable in Splunk SOAR.
`risk_notable_auto_investigate`	This playbook implements an auto-investigate workflow based on a user-defined risk threshold.	A playbook designed to replace `risk_investigate` for organizations looking to adopt a response-first approach. The risk threshold defaults to 250 and should be adjusted as needed.
`risk_mitigate`	This playbook checks for the presence of the Risk Response workbook and updates tasks or leaves generic notes. The `risk_notable_verdict` playbooks recommend this playbook as a second phase of an investigation. You can also use this playbook in ad-hoc investigations or incorporate it into custom workbooks.	To configure this playbook to automatically add notes, see the Playbook outputs section of Use the risk notable playbook pack to investigate a risk notable in Splunk SOAR.
`risk_notable_preprocess`	This playbook prepares a risk notable for investigation by performing these tasks: 1. Ensuring that a risk notable links back to the original notable event with a card pinned to the HUD. 2. Posting a link to the relevant container in a comment field of	For more information, see Deployment steps for using the playbook pack.

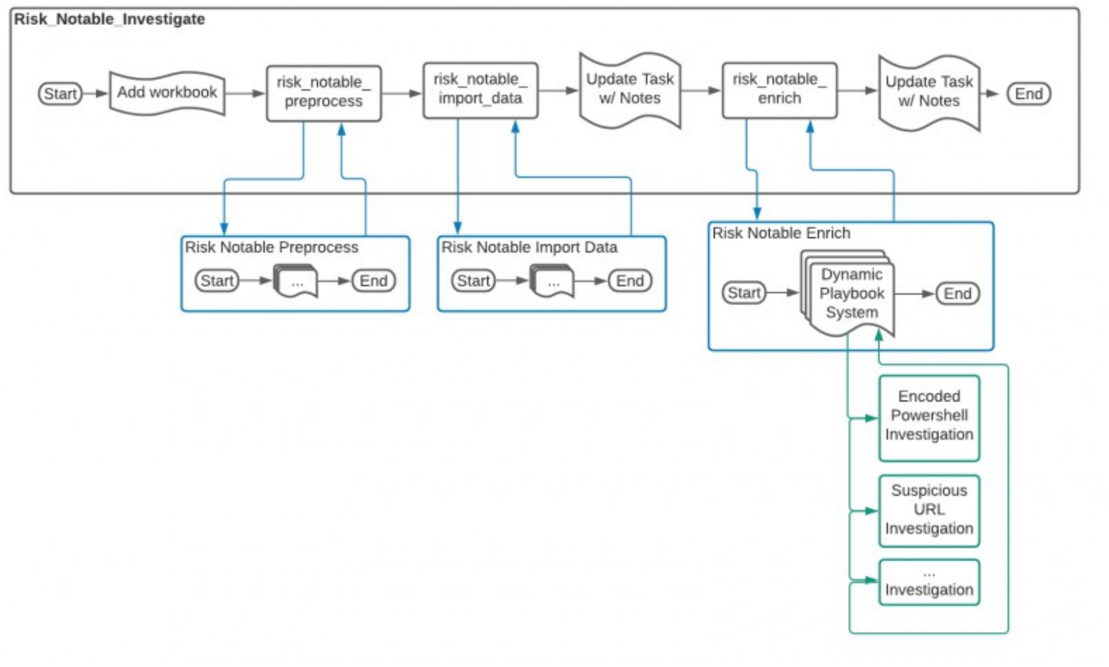
Name	Description	Additional information
	Splunk Enterprise Security. 3. Updating the relevant container's name, description, and severity to reflect the data in the notable artifact.	
risk_notable_import_data	This playbook gathers all of the events associated with the risk notable and imports them as artifacts. It also generates a custom markdown formatted note.	The Splunk search used to locate contributing events requires three fields in the notable artifact: risk_object, iinfo_min_time, and info_max_time. The query also performs some deduplication on contributing events and may need to be adjusted based on individual Enterprise Security environments. Mitre Tactics and Techniques appear if using the annotation framework in Splunk ES. See Use security framework annotations in correlation searches in the *Administer Splunk Enterprise Security* manual. A custom code block sorts the returned event data and produces a markdown formatted note into the note_content output field. This field is then available for use in downstream playbooks.
risk_notable_enrich	This playbook collects the available Indicator data types within the event as well as available investigative playbooks. It will launch any playbooks that meet the filtered criteria.	See Call child playbooks with the dynamic playbook system for more information on building or customizing a playbook for inclusion with risk_notable_enrich.
risk_notable_merge_events	This playbook finds related events based on key fields in a risk notable and allows the user to process the results and decide which events to merge into the current investigation.	Combining the list_merge utility within the playbook with the find_related_containers utility allows for fine-tuning of related event criteria. For example, the default filtering criteria uses description, risk_object, and threat_object as the important fields and requires at least three matches before an event is considered related. There are several options to customize the associated criteria, including adding more fields in list_merge, reducing or increasing the minimum match count, or utilizing the wildcard feature of find_related_containers.
risk_notable_auto_merge	This playbook finds similar or duplicate events based on the risk_object field in a Risk Notable. If two or more events are found with no case, a case will be created with the current container. If a case is found, this container will be merged with the case.	Unlike risk_notable_merge_events, this playbook will not prompt the user before merging. It will only consider events to be similar if they share the exact same value from the field called "risk_object."
risk_notable_verdict	This playbook locates available playbooks with the responses_option tag and presents them to the analyst.	Add response_option to any playbook that should show up in this prompt.

Name	Description	Additional information
	Based on the analyst selection, it will launch its chosen playbook.	
`risk_notable_review_indicators`	This playbook was designed to be called by a user to process indicators that are marked as suspicious within the SOAR platform. Analysts will review indicators in a prompt and mark them as blocked or safe.	See Indicator tagging system for more information about the blocking workflow.
`risk_notable_block_indicators`	This playbook handles locating indicators marked for blocking and determining if any blocking playbooks exist. If there is a match to the appropriate tags in the playbook, a filter block routes the name of the playbook to launch to a code block.	See Call child playbooks with the dynamic playbook system for more information on building or customizing a playbook for inclusion with `risk_notable_protect_assets_and_users`. See Indicator tagging system for more information about the blocking workflow.
`risk_notable_protect_assets_and_users`	This playbook attempts to find assets and users from the notable event and match those with assets and identities from Splunk ES. If a match was found and the user has playbooks available to contain entities, the analyst decides which entities to disable or quarantine.	See Call child playbooks with the dynamic playbook system for more information on building or customizing a playbook for inclusion with `risk_notable_protect_assets_and_users`.
`risk_notable_auto_containment`	Implements an auto-containment of available assets and identities found in artifacts with high risk scores or confirmed threats.	Enable input playbooks that accept entities to be contained such as hosts or users. Adjust artifact filter in import data as needed to select which artifacts are considered to contain entities that should be routed to containment.
`risk_notable_auto_undo_containment`	This playbook gathers contained assets and identities from the container and sends them playbooks with "undo_containment" as well as "asset" or "identity" tags.	Enable input playbooks that are designed to undo the actions performed by containment playbooks.
`reset_entity_risk`	This playbook grabs all of the contributing `risk_rules` in the event that haven't had a risk score reset. It then posts negating risk scores to Splunk after prompting the user for a reason. If no risk rules are present, a comment will be left.	This playbook is designed to be run on individual artifacts or on an entire container. Splunk Enterprise Security administrators may wish to exclude the Splunk SOAR as a source of risk events in Risk Incident Rules. See Use default risk incident rules in Splunk Enterprise Security in Administer Splunk Enterprise Security.

Name	Description	Additional information

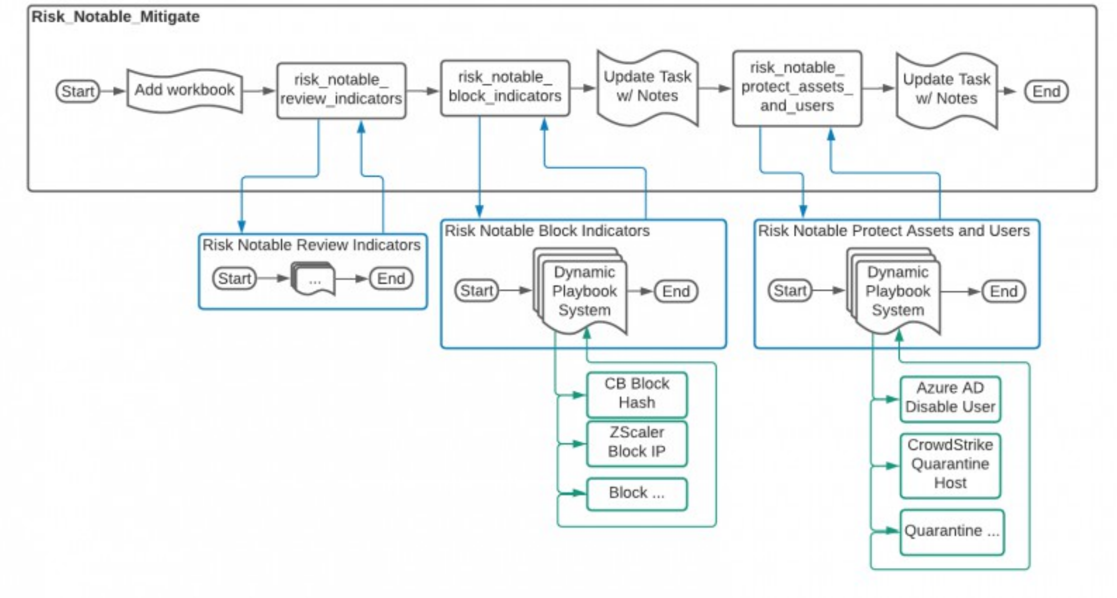
Understand the risk_notable_investigate playbook

The following image shows the workflow of the risk_notable_investigate playbook.



Understand the risk_notable_mitigate playbook

The following image shows the workflow of the risk_notable_mitigate playbook.



Build playbooks compatible with the dispatch_input_playbooks utility

The `dispatch_input_playbooks` utility provides a standardized method for routing data to input playbooks, as well as playbooks in the playbook pack. The utility finds data in the event or container that the utility is being run against, finds registered playbooks, and then routes data to the correct playbooks. After those playbooks finish running, the utility unifies outputs and then passes them downstream.

The `dispatch_input_playbooks` utility is available from the **Playbook** block.

Understand the flow of data through the dispatch_input_playbooks utility

1. Check whether inputs can cause accidental playbook execution.
2. Find playbooks that match the criteria you specify in your input configuration.
3. Find available data types in the specified event or container.
4. Enumerate accepted data types for each playbook and then route data to the correct sub-playbook inputs.
5. Run sub-playbooks.
6. Provide reports for sub-playbook runs, aggregate the resulting data, unify the output, and pass the output downstream.

Inputs

The `dispatch_input_playbooks` utility accepts up to five inputs.

- **playbook_tags**: Access this input from the **find matching playbooks** block. Use this input to find only playbooks that conform to all the tags you provide. You must provide at least one tag or else the dispatcher raises an error.
- **playbook_repo**: Access this input from the **find matching playbooks** block. Use this input to find available playbooks that are located in repositories you provide. If you don't specify a repository, the default is "local." If you try to launch "community" playbooks, an exception is raised.
- **indicator_tags_include**: Access this input from the **collect indicator** block. This input is an enhanced "or" filter

used to make sure indicators with certain tags are included.
- **indicator_tags_exclude**: Access this input from the **collect indicator**block. This input is an enhanced "or" filter used to make sure indicators with certain tags are excluded.
- **artifact_ids_include**: Access this input from the **collect indicator**block. This input is an enhanced filter to make sure artifacts with certain IDs are included.

Outputs

The `dispatch_input_playbooks` has two universal outputs and two situational outputs.

Universal outputs

Universal outputs pass data from the `dispatch_input_playbooks` utility downstream. Universal outputs aren't required. Unless a sub-playbook you dispatch contains universal outputs, they don't exist.

- **note_content**: Use this output to pass downstream a list of formatted strings that you can use to add workbook tasks, comments, or notes. Sample data path: `playbook_dispatch_input_playbooks_1:playbook_output:note_content`.
- **verdict**: Use this output to pass downstream a list of strings representing overall playbook verdicts, instead of any specific indicator verdicts. Sample data path: `playbook_dispatch_input_playbooks_1:playbook_output:verdict`.

Situational outputs

- **sub_playbook_outputs**: Use this output to pass downstream a list of dictionaries representing all outputs generated by sub-playbook runs, organized by `playbook_name`. Each unique playbook represents one entry in the list, which you can filter by. If no dispatched playbooks have outputs, the path and its contents don't exist. Sample data path: `playbook_dispatch_input_playbooks_1:playbook_output:sub_playbook_outputs`.
    - **playbook_name**: This output is a single-value string that represents the human-readable name of the playbook that was launched, not an auto-generated block name. Sample data path: `playbook_dispatch_input_playbooks_1:playbook_output:sub_playbook_outputs.playbook_name`.
    - **<output1>**: A playbook's output displays in the same dictionary where the playbook's name is. You can't filter nested paths beyond this key. Sample data path (filterable): `playbook_dispatch_input_playbooks_1:playbook_output:sub_playbook_outputs.ip_reputation`.

Sample data path (not filterable):
`playbook_dispatch_input_playbooks_1:playbook_output:sub_playbook_outputs.ip_list.*.ip_reputation`

- **sub_playbook_inputs**: Use this output to pass downstream a list of dictionaries that contain all inputs sent to sub-playbook runs, organized by the name of the playbook. Each unique playbook represents one entry in the list, which you can filter by. Unlike **sub_playbook_outputs**, this output exists so long you launch any `input_playbooks`. Sample data path: `playbook_dispatch_input_playbooks_1:playbook_output:sub_playbook_inputs`.
    - **playbook_name**: This output is a single-value string that represents the human-readable name of the playbook that was launched, not an auto-generated block name. Sample data path: `playbook_dispatch_input_playbooks_1:playbook_output:sub_playbook_inputs.playbook_name`.
    - **<input1>**: A playbook's input displays in the same dictionary where the playbook's name is. Sample data path: `playbook_dispatch_input_playbooks_1:playbook_output:sub_playbook_inputs.username`.

Safety measures

The playbook pack has multiple safety measures that ensure the content is performant and doesn't negatively affect Splunk SOAR deployments.

- *Content can't be launched directly from the community*. Prohibiting launches from the community ensures that no content is launched without the SOAR admin or automation engineer's explicit direction, so playbooks must be copied to `local` or another non-community repository before you launch a playbook using this method.
- *Content must be launched with a tag*. Making tags required reduces the likelihood of content being launched without a content author explicitly declaring that an `input_playbook` be used for the intended purpose.
- *Data types must be gathered in a performant way*. The method of gathering playbooks, gathering available data, and routing them has been stress tested against a SOAR environment with thousands of indicators, and thousands of containers being generated per hour.
- *Data types must match accepted input types*. No input playbook that accepts one data type can be given a data type the input playbook isn't able to handle. However, a playbook can receive data that is miscategorized. For that reason, playbook authors should ensure their playbooks handle those situations by filtering or erroring out appropriately.

Use the tagging system with the playbook pack for Splunk SOAR

Tags allow you to call playbooks any time the tag is present.

Use input playbook tags compatible with the playbook pack

Playbooks in specified repositories are automatically called if the associated tag is present. The default repository is `local`.

All input playbooks must include "risk_notable" in addition to the tag itself.

Playbook use	Tags (required)	Outputs (optional)
Investigation or enrichment	`investigate`	`note_title`, `note_content`
Blocking indicators	`block`	N/A
Containment of assets	`asset`, `containment`	N/A
Containment of identities	`identity`, `containment`	N/A
Undo containment of assets	`asset`, `undo_containment`	N/A
Undo containment of identities	`identity`, `undo_containment`	N/A

Understand the indicator tagging system

The `risk_notable_review_indicators` and `risk_notable_block_indicators` playbooks use the `indicator_get_by_tag` utility to fetch indicators with specific tags. To include an indicator with the playbook pack, the playbook used to investigate the indicator type must tag that indicator using the `indicator_tag` utility.

This table lists the available indicator tags and how you can use them:

Indicator tag	How the playbook pack uses the indicator tag	

		How you should use the tag in custom input playbooks
`suspicious` `malicious`	The `risk_notable_review_indicators` playbook alerts the user to any indicators that contain this tag.	When building an investigation playbook, use this tag with an indicator. See the Example child playbook deployment topic for an example of how to deploy a child playbook.
`safe`	The `risk_notable_review_indicators` and `risk_notable_block_indicators` playbooks ignore indicators with this tag.	When building investigation playbooks, use this tag to mark safe indicators.
`marked_for_block`	* The `risk_notable_review_indicators` playbook alerts the user to any indicators that contain this tag. • The `risk_notable_block_indicators` playbook blocks any indicators with this tag.	N/A
`blocked`	* The `risk_notable_review_indicators` playbook ignores indicators with this tag. • The `risk_notable_block_indicators` playbook reports any indicator with this tag, marking each indicator as "successfully blocked."	When building a blocking playbook, use this tag to mark indicators when successful blocks occur.
`known_asset` `known_identity`	* The `risk_notable_auto_containment` playbook routes indicators with this tag to containment input playbooks. • The `splunk_enterprise_security_tag_assets_and_identities` playbook automatically applies this tag.	N/A
`contained`	* The `risk_notable_auto_containment` playbook ignores this tag. • The `risk_notable_undo_containment` playbook routes indicators with this tag to undo containment input playbooks.	When building a containment playbook, use this tag to mark indicators when successful containments occur.