# Table of Contents

## Introduction
- About Splunk Phantom.........................................................................................................................1
- Who should read this manual?..................................................................................................................3
- Log in and navigate Splunk Phantom........................................................................................................3

## Get started using Splunk Phantom
- Start with Investigation in Splunk Phantom..........................................................................................7
- Manage the status, severity, and resolution of events in Splunk Phantom..............................................8
- Approve actions before they run in Splunk Phantom...............................................................................9
- Mark files and events as evidence in Splunk Phantom.............................................................................11
- View recommendations for mission experts, playbooks, and actions......................................................12
- View and create notes in Splunk Phantom...............................................................................................13
- Search within Splunk Phantom...............................................................................................................13
- View the list of configured playbooks in Splunk Phantom........................................................................14
- Create custom lists for use in Splunk Phantom playbooks.......................................................................16

## Use the Visual Playbook Editor to create playbooks in Splunk Phantom
- Create playbooks to automate analyst workflows in Splunk Phantom......................................................18
- Create and debug playbooks in Splunk Phantom using the visual playbook editor...............................18
- Save a playbook so that Splunk Phantom can access it............................................................................31
- View or edit playbook settings................................................................................................................32
- View or edit the Python code in Splunk Phantom playbooks....................................................................33

## Manage cases in Splunk Phantom
- Overview of cases......................................................................................................................................35
- Create cases in Splunk Phantom...............................................................................................................35
- Add objects to a case in Splunk Phantom.................................................................................................36
- Define a workflow in a case using workbooks in Splunk Phantom..........................................................37
- Create case reports to download and share in Splunk Phantom...............................................................39

## Use Splunk Phantom in a Connected Experiences App
- Use the Splunk Mobile app for Splunk Phantom......................................................................................40
- Run Splunk Phantom playbooks in Splunk AR workspaces.....................................................................41

## Use the command line interface to perform tasks in Splunk Phantom
- Splunk Phantom command-line interface overview..................................................................................42
- Run an action in Splunk Phantom............................................................................................................44
- Run a playbook in Splunk Phantom...........................................................................................................45
- Add a note in Splunk Phantom..................................................................................................................45
- Update or edit an event in Splunk Phantom...............................................................................................46
- Use a datapath in Splunk Phantom...........................................................................................................46
Introduction

About Splunk Phantom

Splunk Phantom is a world-class Security Orchestration, Automation, and Response (SOAR) system. The Splunk Phantom platform combines security infrastructure orchestration, playbook automation and case management capabilities to integrate your team, processes, and tools together.

The diagram shows the end-to-end flow of security automation in Splunk Phantom.
The main components of Splunk Phantom each play a role in delivering end-to-end security automation.

<table>
<thead>
<tr>
<th>Component</th>
<th>Description</th>
</tr>
</thead>
</table>
| **App**          | An **App** extends the Splunk Phantom platform by adding connectivity to third party security technologies. The connections allow Splunk Phantom to access and run actions that are provided the third party technologies. Some apps may also provide a visual component such as widgets that can be used to render data produced by the app. This example shows three apps in a Splunk Phantom environment and the actions provided by each app:  
• The MaxMind app provides an action to find the geographical location of an IP address.  
• The PhishTank app provides an action to find the reputation of a URL.  
• The Palo Alto Networks (PAN) Firewall app provides several actions, such as blocking and unblocking access to IP addresses, applications, and URLs.  
See Add and configure apps and assets to provide actions in Splunk Phantom in *Administer Splunk Phantom*. |
| **Asset**        | An **Asset** is a specific instance of an app. Each asset represents a physical or virtual device within your organization such as a server, endpoint, router, or firewall. For example, you have a Palo Alto Network (PAN) firewall app that connects the firewall to Splunk Phantom. Then, you configure an asset with the specific connection details to a specific firewall. If your environment has multiple firewalls, you can configure one asset for each firewall.  
Assets must be configured within Splunk Phantom in order to automate actions on them. Assets typically also possess primary and secondary owners.  
This example shows one MaxMind asset, one PhishTank asset, and two PAN firewall assets. The PAN assets have different version numbers, which is the reason for having two assets.  
See Add and configure apps and assets to provide actions in Splunk Phantom in *Administer Splunk Phantom*. |
| **Container**    | A **container** is a security event that is ingested into Splunk Phantom.  
Containers have the default label of **Events**. Labels are used to group related containers together. For example, containers from the same asset can all have the same label. You can then run a playbook against all containers with the same label.  
You can create custom labels in Splunk Phantom as needed. See Configure labels to apply to containers. |
| **Case**         | A **case** is a special kind of container that can hold other containers. For example, if you have several closely related containers for a security incident, you can promote one of those containers to a case and then add the other related containers to the case. Doing this enables you to consolidate your investigation rather than having to investigate each container individually.  
See Overview of cases. |
| **Artifact**     | An **artifact** is a piece of information added to a container, such as a file hash, IP address, or email header. |
| **Indicator, or Indicator of Compromise (IOC)** | An **indicator** is a piece of data such as an IP address, host name, or file hash that populates the Common Event Format (CEF) fields in an artifact. Indicators are the smallest unit of data that can be acted upon in Splunk Phantom. |
## Component Description

<table>
<thead>
<tr>
<th>Component</th>
<th>Description</th>
</tr>
</thead>
</table>
| **Playbook** | A **Playbook** defines a series of automation tasks that act on new data entering Splunk Phantom. For example, you configure a playbook to run specific against all new containers with a specific label. Or, you can configure running a playbook as part of the workflow in a workbook. In this example, two playbooks are configured:  
  - **Playbook 1** runs the following actions whenever a new container is created in Splunk Phantom:  
    - The geolocate ip action, using the MaxMind asset.  
    - The block ip action, using the version 2.7 PAN Firewall asset  
  - **Playbook 2** runs the following actions whenever a specific workbook is used in a case:  
    - The url reputation action from the PhishTank asset.  
    - The list application, unblock application, and unblock URL actions using the version 3.0 PAN Firewall asset.  

  See [Create playbooks to automate analyst workflows in Splunk Phantom](#). |
| **Workbook** | A **workbook** is a template providing a list of standard tasks that analysts can follow when evaluating containers or cases.  

  See [Define a workflow in a case using workbooks in Splunk Phantom](#). |
| **Action** | An **Action** is a high level primitive used throughout the Splunk Phantom platform, such as **get process dump**, **block ip**, **suspend vm**, or **terminate process**. Actions are run in playbooks or manually from the Splunk Phantom web interface.  

  Actions are made available to Splunk Phantom by apps. See [Add and configure apps and assets to provide actions in Splunk Phantom in *Administer Splunk Phantom*](#). |
| **Owner** | An **Owner** is responsible managing assets in your organization. Owners receive approvals, which are requests to execute a particular action on a particular asset. Approvals are sent to the asset owners and contain a service level agreement (SLA) dictating the expected response time. Approvals are first sent to the primary asset owners. If the SLA is breached, then the approval is redirected to the secondary asset owner. SLAs can be set on events, phases, and tasks.  

  - See [Configure approval settings for a Splunk Phantom asset in *Administer Splunk Phantom*](#).  
  - See [Configure the response times for service level agreements in *Administer Splunk Phantom*](#) for more information about configuring SLAs. |

### Who should read this manual?

This manual is intended for Security Operations Center (SOC) staff, analysts, and managers who are not primarily Splunk Phantom administrators.

### Log in and navigate Splunk Phantom

The Phantom web interface requires a browser with HTML 5, SVG graphics and current TLS support. See [Supported browsers in *Install and upgrade Splunk Phantom*](#).

To access the Phantom web interface, perform the following tasks:
1. Enter the IP address you configured for your virtual appliance, or the DNS name if you created one for the IP address.
2. Enter your login credentials. The default administrative user is **admin** and the password is **password**.

The Splunk Phantom home page shows graphs and statistics that are useful for users managing incidents and actions.

**Access Account Settings**

Click your account name and select **Account Settings** to access your account settings.

The default admin account on a Splunk Phantom instance is a local account. Local accounts only exist in the database for the Splunk Phantom web interface and can’t be used to log into the operating system or any external authentication server.

Each account must have at least one email address associated with it. Splunk Phantom uses this email address as part of the approval process workflow.

Splunk Phantom also supports single sign-on authentication from various identity providers. For more information, see Configuring single sign-on authentication for Splunk Phantom in the *Administer Splunk Phantom* manual.

**Account Settings**

You can configure various settings through the account settings page. Use this page to configure user settings, notifications, change your password, and register a mobile device.

**User Settings**

For LDAP users, First Name, Last Name, Title, and Location are pulled from LDAP and automatically filled in, so you cannot edit those. The email field is also pulled from LDAP if it exists, but it’s editable in case you want to receive email elsewhere, and because the email field in Phantom is critical for sending email notifications.

For a local account, the primary email is the username you log in with. You can change it at any time, but you must use the new email address the next time you log in. Your current login session continues until you log out, your session expires, or you switch browsers or machines.

**Notifications**

You receive email notifications when the status of an incident changes or when an incident is about to expire. You can configure your notification settings to meet your needs.

You can view Splunk Phantom notifications on your mobile device using the Splunk Mobile app. See View a notification to learn how to view Splunk Phantom notifications on your mobile device.

Some notification types are not implemented. For example, there is not yet an incident watchlist, so the "Incident on watchlist changed" notification does not do anything in the current version of Splunk Phantom. Also, privileges have not yet been implemented on notifications, so users with no particular roles can opt to see notifications for All Incidents.

**Change Password**

Only local users will see the Change Password tab. LDAP users will need to change their password on the Active Directory or other LDAP server that they are using to log into Phantom. For a local user to change their password, they
enter their current password in the appropriate field, and enter the new password twice (since it will be showing asterisks instead of what they typed), and then they click the Change Password button. Password complexity settings may be adjusted in the Account Security tab under User Management (Administration section).

**Mobile device registration**

Contact a Splunk Phantom admin to enable mobile device registration so that you can register your mobile device and get started with the Splunk mobile apps. You can register multiple devices, but one device cannot be registered to multiple users.

**Prerequisite**

An admin must enable the mobile feature on your Phantom instance. See Enable mobile device registration with Splunk Phantom in the *Administer Splunk Phantom* manual.

**Steps**

1. Download Splunk Mobile onto your mobile device and install it.
   See Download Splunk Mobile for iOS or Download Splunk Mobile for Android to download the Splunk Mobile app.
2. On your mobile device, tap the gear icon to get to the Settings page.
3. Tap **manage instances** in the Settings page.
   A pop-up opens.
4. Tap the word **edit** in the pop-up.
5. Tap the **Register** button in the pop-up. The 10-digit code appears on the mobile device.
6. From your Splunk Phantom account settings menu in the Splunk Phantom instance click on **Name > Account Settings > Mobile Device Registration**:
   1. Enter the 10-digit code from the mobile device into the activation code fields.
   2. Enter a name for the device in the device name field.
   3. Click **Register**.
   4. A confirmation code pop-up window appears. If this matches the confirmation code on the mobile device, then login with your Splunk Phantom instance credentials, and click **Continue**.
   7. To add additional devices, click **+ New Device** and follow the previous steps again.

Your registered devices display in a table. These are the devices that will receive push notifications from Splunk Phantom for approvals, prompts, manual tasks, and workbook tasks.

When you no longer want access to the mobile app from a particular mobile device, you can unregister from the app itself or you can do the following to unregister the device on Splunk Phantom:

1. Locate the device by name or by type in the table.
2. In the Action column, click **remove**.
3. Confirm at the prompt by clicking **remove**.

It is also possible for a Splunk Phantom admin to unregister your mobile device. If an admin unregisters your device, it will disappear from your account settings. If an admin deletes your user account, you will no longer be able to access Splunk Phantom through the mobile app.

**View Splunk Phantom product documentation**

Users can set a default choice for viewing documentation. The default is Online, which means that the documentation links in the platform go to docs.splunk.com. Offline means that the documentation links in the platform go to a local copy
of the docs in PDF format.

Online docs are continuously updated and include the most recent changes and improvements to the Splunk Phantom documentation. You can create an offline copy of the docs at any time using the **Download manual as PDF** link on the page at docs.splunk.com.

To change the default view to Offline, complete the following steps:

1. Navigate to the main menu.
2. Select **Documentation**.
3. Click **Offline**.
Get started using Splunk Phantom

Start with Investigation in Splunk Phantom

Use the Splunk Phantom Investigation page as the starting point to understand, investigate, and act on events. Investigation provides you access to event activity history, contextual and interactive data views, secure file attachments, and automation and case management controls.

The activity feed displays current and historical action and playbook activity that has acted on the currently displayed event. It provides a summary of the success, ongoing execution, and results of all automation operations for the event. The activity feed also provides team collaboration capabilities that are integrated inline with automation details and other data, forming a record of all relevant event information.

You can use Splunk Phantom to promote a verified event to a case using the integrated case management capability. Case management supports tasks that map to your defined Standard Operating Procedures (SOPs). Case management also has full access to the Phantom Automation Engine, allowing you to launch actions and playbooks as part of a task.

Set your view in Investigation

Analyst and summary views enable different personas to quickly view information and perform actions. Toggle quickly between the summary and analyst views by clicking the Summary or Analyst view buttons in an event or case.

- The Summary view presents mostly non-actionable information about a case. This information is useful for individuals such as managers or executives who want to be able to view the status of a case without having to view the actionable items.
- The Analyst view contains the same information as the summary view along with all options to perform actions on the case, such as run a playbook, add and edit a workbook, or view and add artifacts.

HUD cards

The collapsible heads up display (HUD) helps you track important metrics and information. Splunk Phantom administrators control HUD card settings. Users can customize the HUD for an event or case by adding or removing cards, or configuring manual cards of their own design.

The following HUD card types are available:

- Preset Metrics
- Custom Fields
- Manual

Preset Metrics and Custom Fields cards are defined by a Splunk Phantom administrator and display one of the built-in metrics or the information from a custom field. You can add or remove these cards, but only an administrator can change the card options. Manual cards let you add a customized card to the HUD for an event or case.

Add a card to the HUD

Perform the following steps to add a card to the HUD:

1. From the Phantom main menu, select either Cases or Sources > My Events.
2. Select an event or case.
3. Expand the HUD menu.
4. Click the gear icon to open the Configure HUD modal.
5. Click + HUD Card.
6. Choose a HUD card type.
7. Configure the available card options. The following table describes the manual card options:

<table>
<thead>
<tr>
<th>Setting</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>Text creates an input field where you can add a small amount of text. Select creates a card with a dropdown list of options.</td>
</tr>
<tr>
<td>Message</td>
<td>The name of the HUD card.</td>
</tr>
<tr>
<td>Color</td>
<td>The display color of the HUD card.</td>
</tr>
</tbody>
</table>

8. Click Save.

To display HUD information from earlier versions of Splunk Phantom, set HUD TABLE DATA to ON.

Manage the status, severity, and resolution of events in Splunk Phantom

You can manage the status, severity, and resolution of events in Splunk Phantom in order to best organize events.

Use status to represent the state of an event

Each event or case has a status. Use the status to indicate the state of an event or case.

Statuses are grouped into three types: New, Open, and Closed. You can create up to 10 additional custom statuses in each category as required by your business processes.

The status of an event or case is set when it is created or ingested from an asset.

Perform the following steps to change the status of an event or case:

1. In Investigation, click the downward arrow stack icon next to the Playbook button.
2. In the expanded section at the top of the page, click Event Info.
3. Select a status from the menu in the Status field.

You can also set the status of a case or event using actions inside of a playbook. See Set container parameters using the API block.

Use severity to represent the importance of an event

Severity defines the impact or importance of an event or case. Different severities have their own service level agreements (SLAs) assigned to them.

Splunk Phantom ships with three severity names: High, Medium, and Low. Your organization might need additional levels of severity to match your business processes. A Splunk Phantom administrator can define additional severity names.

The severity of a case or event is set when it is created or ingested. You can change the severity assigned to a case or event in Investigation by clicking on the severity label.
Each severity label has a corresponding SLA which is defined as the number of minutes that can pass before an action or approval is considered late. Each severity name can be configured with its own SLA.

This table lists the default SLA settings for High, Medium, and Low.

<table>
<thead>
<tr>
<th>Severity name</th>
<th>SLA</th>
</tr>
</thead>
<tbody>
<tr>
<td>High</td>
<td>60 minutes (1 hour)</td>
</tr>
<tr>
<td>Medium</td>
<td>720 minutes (12 hours)</td>
</tr>
<tr>
<td>Low</td>
<td>1440 (24 hours)</td>
</tr>
</tbody>
</table>

Use SLAs for the following purposes in Splunk Phantom:

- Track the amount of time an event or case has remaining before it is considered due.
- Track the amount of time an approver has to approve an action before the approval is escalated to another approver.

If an approver does not approve an action before the SLA time elapses, the action is escalated to the next level of approvers.

For more information about the approval and escalation process see Approve actions before they run in Splunk Phantom.

**Close or resolve events and cases**

When all the tasks or actions associated with a case or event are complete, you can close or resolve the case or event by setting the status to a Closed type. You can change the status in Investigation, using the REST API, or by automation in a playbook.

Change the status of an event or case by selecting the status from the menu in Investigation > Event Info > Status. Playbooks can also set the status of a case or event.

An administrator can specify which tags are required before an event or case before you can resolve it. Selecting a status with a Closed type with a missing required tag generates an error.

**Approve actions before they run in Splunk Phantom**

Take action on an asset to either make it do something or retrieve information from it. For example, you can create an action to use a firewall to block a particular IP address, request a list of VMs from a VMware ESXi server, or look up a file hash on VirusTotal.

Action approval is controlled at the asset level. You can assign an asset to one or more approvers. If someone takes action on that asset, all approvers must approve the action before it runs. If an asset has no approvers, or if the actions are read-only, all actions taken on it run immediately. A read-only action is an action that does not change anything on the device or application with which it is communicating. For example, a read-only action from a firewall obtains information from the firewall without doing anything to change the firewall.

**Take action on assets**

An asset doesn’t need to have approvers. Approval is only required for actions that have a write component.
To start an action, perform the following steps:

1. From the main menu, click **Sources**.
2. Select a label.
3. Select an event.
4. Click **Analyst** to make sure you are in analyst view.
5. Click **Action**.

For more information about how to assign approvers to assets, see Add and configure apps and assets to provide actions in Splunk Phantom in the *Administer Splunk Phantom* manual.

The following diagram describes the approval escalation path in Splunk Phantom:

```
Primary approvers  Secondary approvers  Executive approvers
```

**Primary approvers**

If an asset has primary approvers, the required number of approvers must approve the action within the action service level agreement (SLA) deadline. If any single primary approver denies the action, the action stops immediately and no further approvals are permitted.

**Secondary approvers**

If the minimum number of approvals by primary approvers isn't met within the SLA, but no one denied the action, it moves to secondary approvers and the action SLA clock starts over.

**Executive approvers**

If the action isn't approved or denied within the secondary SLA, it moves to the executive approvers. Executive approvers have the same amount of time to approve the action. If they fail to act on it, it expires and doesn't run.

**Run actions using the Splunk Phantom API**

You can also run actions using a call to the phantom.act API. See act in the *Splunk Phantom Playbook API Reference* manual. The *reviewer* parameter is optional. If you specify this parameter, the action doesn't run until a reviewer approves or denies it. A reviewer is an analyst or member of the security operations team who reviews and decides if the action is allowed to run. After the reviewer approves the action, the approval process begins.

The first reviewer to approve or deny the action determines whether it runs or not. If the SLA expires before any reviewer approves it, the action fails.

**Delegate actions to other users**

When an approver receives a notification to approve an action, they can delegate the approval to one or more users or roles. Those users must approve or deny the action within the remaining SLA period or it moves to the next level of
approval.

If you delegate an action, you renounce your portion of the vote. All delegates must approve the action for it to count as a single vote from the original approver.

**Delegation example**

Users A, B, and C are primary approvers for an asset that requires two approvers. Users D and E are secondary approvers, and they all need to approve the action. Users F, G, H, and I also exist on the system. Someone takes an action on the asset and users A, B and C are notified.

If any two users approve the action within the SLA deadline, the action runs. Alternatively, user A might approve while user B delegates to users F and G. The action runs if users F and G both approve the action. While A approved directly, B effectively approved by being represented by F and G. The single-user veto still applies because users F or G can deny the action.

Alternatively, if users F or G don't approve but C does, the action runs because A and C make up the two approval votes needed.

A second level of delegation is allowed. When user B delegates to F and G, user F can then delegate to users H and I. The requirement that all delegates must approve still stands. To represent user B's original vote, G, H, and I must all approve. G has half a vote, and H and I each have a quarter of a vote.

**Delegation restrictions**

You can't delegate to other approvers that an action is currently waiting on. Primary approvers can delegate to secondary approvers, but not other primary approvers. You can delegate to primary approvers only once an action expires and moves to the secondary approvers.

**Mark files and events as evidence in Splunk Phantom**

When you discover information that's critical to your conclusions, you can mark it as evidence in your investigation. Evidence can include files, artifacts, action results, events, and notes. Tagging evidence helps to separate general information from information that's directly related to diagnosing an incident. All evidence appears on the Evidence tab within an investigation.

**Mark a file as evidence**

Perform the following steps to mark a file as evidence:

1. Navigate to a container in Splunk Phantom.
2. Click Analyst to change the container to analyst view.
3. Click the Files tab.
4. Click the ... button for the file.
5. Select Mark as Evidence.
6. Click Confirm.
Mark an event as evidence

To mark an event as evidence, select **Mark as Evidence** when adding the event to a case. For more information about adding events to cases, see *Add objects to a case in Splunk Phantom*. The following screenshot illustrates how to mark an event as evidence.

When you add an event to an existing case, it copies all the data from the existing event into the case you're adding it to while also maintaining the original event data. The information on the *Evidence* tab is a copy of the original event, not the actual event.

View recommendations for mission experts, playbooks, and actions

Use the *Guidance* tab to view recommended users, playbooks, and actions that can be used to resolve an event. The recommendations are provided by Splunk Phantom based on a variety of factors, including the following:

- Previous playbooks or actions run on a container or case with the same label.
- The users working on that label.
- The frequency with which those previous entities were used. For example, a user that has frequently changed the state of all containers with the matching label would be considered an expert.
- How recently an entity has interacted with the case or container. For example, a user is considered less of an expert as time goes on, assuming there is no activity from the user.

Perform the following tasks to view guidance information:

1. Navigate to a container or case in Splunk Phantom.
2. Click **Analyst** to switch to Analyst view.
3. Click the **Guidance** tab.

The *Mission Experts* are the users who have taken action on containers or cases with the same label. You can also view recommended *Playbooks* and *Actions* in their respective sections.
View and create notes in Splunk Phantom

You can create a note in Splunk Phantom when working with tasks, artifacts, or cases. Use the Notes tab to view all of the notes, regardless of the user who created them. For example, perform the following tasks to view all notes for a container or case:

1. Navigate to a container or case in Splunk Phantom.
2. Click the Notes tab.

You can create a general note by clicking + Note.

You can filter the notes being shown by doing the following:

- In the Show field, select Task Notes, General Notes, or Artifact Notes from the drop-down list. By default, all notes are displayed.
- In the Phase field, select a workbook phase from the drop-down list.

Users who upgrade from releases of Splunk Phantom earlier than version 4.5 need to re-index containers before notes are searchable. Use the Search Settings page to re-index.

Search within Splunk Phantom

Splunk Phantom includes an embedded copy of Splunk Enterprise for searching data in the Splunk Phantom instance. You can also configure search using an external instance of Splunk Enterprise or external Elastic Search. For more information, see Configure search in Splunk Phantom in the Administer Splunk Phantom manual.

The search terms appear as part of the URL in the address bar, so you can create a bookmark using the search terms. For example:


Each time the page loads, the search results might vary as changes in Splunk Phantom occur between page visits.

When no filters are selected, an implied ALL condition applies to the search. Use the filters such as Containers, Artifacts, or Actions to narrow your search results. When filters are selected, any categories not selected are excluded from the search.

By default, the search returns 10 results per page. Use the menu to view a maximum of 100 results per page.

Searching with multiple words creates an implied ALL condition. For example, the term data path returns results containing both data and path. Use OR to find results containing either data or path, as shown in the following example:
The search directives in Splunk Phantom are limited to a small subset of the Splunk Processing Language (SPL). If you're using an external Splunk Enterprise instance as your Phantom search engine, you can use all of the Splunk Enterprise features through the interface on that instance. For more information, see Understanding SPL syntax in the Splunk Enterprise Search Reference manual.

The basic boolean operators are AND, OR, and NOT. Use parentheses to group terms and build more complex boolean searches.

The NOT operator excludes an entire object from appearing in the search results, even if other terms do match within that object.

Include quotes for an exact phrase. For example, you can use `data AND path` to explicitly search for objects with both data and path, but you can also use "data path" with quotes to search for that exact phrase. The query returns the word `data` followed by whitespace followed by the exact word `path`.

Search works on whole words, which are strings of non-special characters without whitespace. For example, searching for `data` finds one set of results, while searching for `dat` finds different ones, unless the object happens to contain both the words `data` and `dat`.

Use wildcards to search for partial words. A single asterisk matches any number of characters. For example, searching for `dat*` matches either `data` or `dat` or any words starting with the characters `dat`, such as `date`.

**View the list of configured playbooks in Splunk Phantom**

The playbooks list contains all your currently available Splunk Phantom playbooks and significant metadata about those playbooks. Use the playbooks list to sort, filter, and manage your playbooks.

To open the playbooks list, perform the following steps:
1. From the main menu, select **Playbooks**.
2. Click the **Playbooks** tab if it’s not already open.
3. (Optional) Use the search field to find specific playbooks. Searches are case-insensitive and partial-word matches are supported. This search does not support booleans, such as AND, NOT, or OR.

Use the buttons to reorder the playbooks on this page, configure source control, import playbooks, or create new playbooks:

<table>
<thead>
<tr>
<th>Button</th>
<th>Description</th>
</tr>
</thead>
</table>
| ![Set the order to run playbooks with a status of Active.](image) | Set the order to run playbooks with a status of **Active**.  
- Playbooks with a status of **Inactive** are not run. When you change a playbook’s status to **Inactive**, you are prompted to cancel the running playbook.  
- The next playbook in the list starts once the preceding playbook’s `on_start()` function has completed.  
- If you want one playbook to depend on another playbook finishing completely before starting, use the `phantom.playbook()` function instead of the playbook list. See playbook in the *Splunk Phantom Playbook API Reference*. |
| ![Splunk Phantom stores playbooks in Git repositories. See Configure a source code repository for your playbooks in \*Administer Splunk Phantom\*. Click this button to open the \*Update from Source Control\* dialog.](image) | Splunk Phantom stores playbooks in Git repositories. See **Configure a source code repository for your playbooks in Administer Splunk Phantom**. Click this button to open the **Update from Source Control** dialog.  
1. Select a repository from the drop-down list in the **Source to update from** field.  
2. Select either **Force Update** or **Preserve State**  
   - **Force Update** treats the remote repository as authoritative. Using this overwrites any local changes to playbooks.  
   - **Preserve State** retains the local metadata for changes to playbooks. Playbooks from the community repository always have a status of **Inactive**. If you have set the status of a community playbook to **Active** locally, updating from the community repository will set its status to **Inactive** unless you select **Preserve State**.  
3. Click **Update**. |
| ![Manage source control settings. See Configure a source code repository for your Splunk Phantom playbooks in \*Administer Splunk Phantom\*.](image) | Manage source control settings. See **Configure a source code repository for your Splunk Phantom playbooks in Administer Splunk Phantom**. |
| ![Import a playbook that was exported from another instance of Splunk Phantom.](image) | Import a playbook that was exported from another instance of Splunk Phantom.  
1. Click this button to import a playbook.  
2. In the **Source to update** field, select a repository where you want to write the imported playbook.  
3. (Optional) Click **Force Update** to overwrite existing versions of the same playbook.  
4. Drag and drop a compressed playbook in `.tgz` format, or click and navigate to the playbook.  
5. Click **Upload**. |
| ![Open the Visual Playbook Editor (VPE) to create a new playbook. See \*Use the visual playbook editor to create and debug playbooks in Splunk Phantom\*.](image) | Open the Visual Playbook Editor (VPE) to create a new playbook. See **Use the visual playbook editor to create and debug playbooks in Splunk Phantom**. |

Click the vertical ellipsis (?) icon to toggle the display of the available columns in the playbook list. Items marked with a check mark (?) are displayed in the playbook list. When the space required to display the columns exceeds the width of the current window, a scroll bar appears at the bottom of the playbook list.

**Edit, delete, export, or copy a playbook**

Click the name of a playbook to open it in the Visual Playbook Editor. For more information, see **Use the visual playbook editor to create and debug playbooks in Splunk Phantom**.

Check the checkbox next to the playbook name to select one or more playbooks. After playbooks are selected, you can perform the following actions:
<table>
<thead>
<tr>
<th>Button</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>Edit</td>
<td>Set the properties of the selected playbooks, not the playbooks themselves. Set the status, logging mode, safe mode, which labels the playbook operates on, the category, and tags by selecting the property value you want from the drop-down list.</td>
</tr>
<tr>
<td>Delete</td>
<td>Delete the selected playbooks. A dialog box asks you to confirm your choice.</td>
</tr>
<tr>
<td>Export</td>
<td>Download the playbook as a .tgz extension archive. You can export only one playbook at a time.</td>
</tr>
<tr>
<td>Copy</td>
<td>Save the playbook to a repository that you have configured, such as Git. You can only copy one playbook at a time.</td>
</tr>
</tbody>
</table>

Create custom lists for use in Splunk Phantom playbooks

A custom list is a collection of values that you can use in a Splunk Phantom playbook, such as a list of banned countries, or denied or allowed IP addresses. In your Filter and Decision blocks, compare parameters against all the values in a custom list, rather than having to configure each comparison in the playbook.

Create a custom list in Splunk Phantom

Perform the following steps to create a custom list:

1. From the Phantom main menu, select Playbooks.
2. Click the Custom Lists tab.
3. Click + List to create a new list.
4. Enter a name for the list.
5. Enter the list values in the table using one value per line. For example, you can create a list of banned countries, or denied or allowed IP addresses.
6. Click Save Changes.

See Example of using a custom list in a filter for an example of how to use a custom list in a playbook.

Create a custom list using the REST API

See REST Lists in the Splunk Phantom REST API Reference for information about how to manage custom lists using the REST API.

Export a custom list for use with third party products and services

You can use the REST API to export a custom list for use as an external deny list with third party products and services. For example, you can publish a list of banned IP addresses that can be used in your Palo Alto Networks firewall products.

Perform the following tasks to export a Splunk Phantom custom list and use it in a third party product.

1. Review the formatting requirements that your third party product or service has for custom lists. For example, Palo Alto Networks products may have specific formatting requirements for their dynamic lists. Review these requirements so that the formatting in your Splunk Phantom custom lists match these formatting requirements of your third party product or service.
2. Provide a URI to the custom list in Splunk Phantom using the following format:

   https://username:password@[phantom server]/rest/decided_list/[list name]/formatted_content?_output_format=csv

   For example, to provide a URI to the Splunk Phantom server phantomserver.example.com, using admin as the user and password as the password, and a custom list named blockdomains:
https://admin:password@phantomserver.example.com/rest/decided_list/blockdomains/formatted_content?_output_format=csv
Use the Visual Playbook Editor to create playbooks in Splunk Phantom

Create playbooks to automate analyst workflows in Splunk Phantom

Create a playbook in Splunk Phantom to automate security workflows so that analysts can spend more time performing analysis and investigation. The visual playbook editor (VPE) provides a visual platform for creating playbooks without having to write code.

To define a workflow that you want to automate, link together a series of actions that are provided by apps. An app is third-party software integrated with Splunk Phantom. For example, you can integrate MaxMind as a connector, which provides a geolocate ip action, or integrate Okta as a connector to provide actions such as set password or enable user. The actions available for use in your playbooks are determined by the apps integrated with Splunk Phantom.

After you create and save a playbook in Splunk Phantom, you can run playbooks when performing these tasks in Splunk Phantom:

- Triaging or investigating cases as an analyst
- Creating or adding a case to Investigation
- Configuring playbooks to run automatically directly from the playbook editor

Create and debug playbooks in Splunk Phantom using the visual playbook editor

Perform the following tasks to open the visual playbook editor (VPE) and create a new playbook in Splunk Phantom:

1. From the Main Menu, select Playbooks.
2. Click + Playbook. The VPE opens in a new tab in your browser. The Start and End blocks are populated on the editor. All playbooks must start with the Start block and end with the End block.
3. Specify a name for the playbook.

Playbooks in the same folder cannot have the same name. Playbooks in different folders can have the same name. Click the plus and minus icons to zoom in or zoom out.

Add a new block to the playbook

To add a new block to a playbook, drag the half-circle icon attached to any block on the canvas. Release your mouse to create a new empty block connected to the originating block with an arrow.

When you place a new block on the editor, a set of playbook types appears for you to select:

<table>
<thead>
<tr>
<th>Playbook type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Action</td>
<td>Run an action provided by an app that is installed and configured in Splunk Phantom. For example, you can use the MaxMind connector to geolocate an IP address.</td>
</tr>
<tr>
<td>Playbook type</td>
<td>Description</td>
</tr>
<tr>
<td>---------------</td>
<td>-------------</td>
</tr>
<tr>
<td>Playbook</td>
<td>Run an existing playbook inside your current playbook.</td>
</tr>
<tr>
<td>API</td>
<td>Perform an action by making an API call.</td>
</tr>
<tr>
<td>Filter</td>
<td>Filter the results of the previous block. For example, you can separate items that have a specific severity and perform a different set of actions on those items.</td>
</tr>
<tr>
<td>Decision</td>
<td>Make a decision and perform different actions depending on the results of the previous block. For example, you can exclude all destination IPs that belong to a specific country.</td>
</tr>
<tr>
<td>Format</td>
<td>Format the results of the previous block. For example, you can gather data, format that data in a specific way, and send an email.</td>
</tr>
<tr>
<td>Prompt</td>
<td>Require a user to take action before proceeding to the next block.</td>
</tr>
<tr>
<td>Manual Task</td>
<td>Send a message to a Splunk Phantom user or group that must be acknowledged.</td>
</tr>
<tr>
<td>Custom Function</td>
<td>Add custom Python code to your playbook to expand the kinds of processing that are performed by the playbook.</td>
</tr>
</tbody>
</table>

Add an Action block to a playbook

Perform the following steps to add an Action block to a playbook:

1. Drag the half-circle icon attached to any existing block in the editor.
2. Select **Action** from the list of block types. Actions available to you in the playbook editor are determined by the apps that are installed and configured in Splunk Phantom.
3. Select the action you want to configure, or enter an action name in the search field if you don't see the desired action listed. You can also filter the list of actions by action type.
4. Select **investigate**, **generic**, **correct**, or **contain**.
5. Click **By App** to view a list of configured apps, and select an available action provided by the selected app.
6. Select an asset that you want to run the action on. An asset is a specific configuration or instance of an app. In some cases, you may have multiple configurations for a specific app. For example, your environment may have multiple networks separated by firewalls, which require you to configure one instance of a specific app for each network.
7. Select the field where you want to perform the asset. For example, an IPS event may have fields like **sourceAddress** and **destinationAddress** and the attack signature. When a container is created in Splunk Phantom, it has an artifact with fields for the **sourceAddress** and **destinationAddress** from the event.
8. Select one of these fields to perform the action on.
9. Click **Save**.
10. Enter a comment about this action.

Configure linked parameters

Configure linked parameters in an Action block when you have multiple assets that share parameters with the same name. For example, you might have multiple assets configured that provide an action to create a ticket with a subject parameter. In this case, the word "linked" appears above the subject field, indicating that the field is linked to another field with the same name in a different asset. If you change the value here, the value for the field changes in all assets.

If you need to have the field take separate values, create separate action blocks.
**Advanced settings**

Follow these steps to configure advanced settings for an **Action** block:

1. Click **Advanced Settings**.
2. Select **General Settings**, **Action Settings**, or **Join Settings**.

<table>
<thead>
<tr>
<th>Setting</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>General Settings</td>
<td>Configure settings for this <strong>Action</strong> block.</td>
</tr>
<tr>
<td></td>
<td>• Custom Name: The name for this action block. This name is visible in the playbook editor and also in Splunk Phantom wherever details about this action are visible.</td>
</tr>
<tr>
<td></td>
<td>• Description: The <strong>Description</strong> field shows up as a code comment above the block definition.</td>
</tr>
<tr>
<td></td>
<td>• Notes: The <strong>Notes</strong> field contents appear when you hover over the Note icon in the action block.</td>
</tr>
<tr>
<td>Action Settings</td>
<td>Configure the action settings that a user must perform.</td>
</tr>
<tr>
<td></td>
<td>• Reviewer: Select a user or group that must approve this action before the action runs. If you select a group or role, any user in that role can approve the action.</td>
</tr>
<tr>
<td></td>
<td>• Delay Timer: Set a delay in minutes before the action runs. A clock icon is visible on the action block to show that a delay is configured.</td>
</tr>
<tr>
<td>Join Settings</td>
<td>You can configure Join settings when you have two blocks with callbacks both calling the same downstream block. Block types with callbacks are <strong>Action</strong> and <strong>Prompt</strong>. Configure Join settings from the downstream block. Click the <strong>required</strong> checkbox if the action in the upstream block must be completed before this downstream block is run.</td>
</tr>
</tbody>
</table>

**Use filters to separate artifacts before further processing**

Create conditions in a **Filter** block to separate a subset of artifacts. Only the artifacts matching the specified condition are passed along to downstream blocks for processing. This is useful when you want to remove artifacts that aren't needed in the flow if the playbook, or you need to separate artifacts because they require different blocks for processing. For example:

- If an IP address comes from North Korea or Turkey, you can block it.
- If an IP address comes from North America, you can perform an IP reputation lookup.
- If an IP address falls in the 192.168.10.* range, you can to grant access to the user.

Options in a filter aren't related to each other and you can perform multiple actions on an IP address. For example, both the second and third conditions in the example could be true, as an IP address in the 192.168.10.* range could both come from North America and be an internal user who was granted access. The filtered data set is given a named result in the format `name="filter_1:condition_1"` and passed to the next block for processing.

**Create a Filter block in your playbook**

To create a **Filter** block, perform the following tasks:

1. Create a new block in the visual playbook editor (VPE).
2. Select **Filter** from the list of block types.
3. Click the **Select Parameter** field and select the parameter you want to filter on. Parameters are made available to the **Filter** block by upstream blocks.
4. Click in the `==` field and select an operator for the filter.
5. Click the **Select Value** field and select the value you want to match. See Example of creating a filter for an example of how these fields work together.
6. (Optional) Click `+` to add parameters to this condition.
7. (Optional) Click Add Condition to create another matching condition for the filter. You can have a maximum of five conditions per Filter block. Each condition has its own downstream path.

**Example of creating a filter**

In the following example, you can create a filter to perform a geolocate ip action on a source IP address and block any IP addresses from North Korea. The following image shows the VPE in Splunk Phantom.

1. Configure a geolocate ip action in the playbook editor.
2. Drag the half-circle icon on the geolocate ip action block in the editor.
3. Select Filter from the list of block types.
4. Click the Select Parameter field and select geolocate_ip_1.
5. Select geolocate_ip_1:action_result_data.*.country_name.
6. Leave == as the operator, and type North Korea in the Select Value field.

**Example of creating a filter with multiple conditions**

You can create multiple rows within a condition or multiple conditions. The following image shows an example of the VPE in Splunk Phantom.
1. Click **Add Condition** to create a second set of filter conditions, which also adds a second output point on the **Filter** block.
2. In the **Select Parameter** field, select **container properties** and choose **label**.
3. Select **==** as the operator, and enter **Test** in the **Select Value** field.

Multiple conditions within a filter block are independent of each other. The results of Condition 1 don’t play into the set of inputs for Condition 2. In this example, Condition 1 uses the result from the **geolocate ip** action, while Condition 2 uses a property of the container. Each condition has its own color to make it easier to identify the separate downstream actions. Green marks the path of Condition 1 and purple marks the path of Condition 2.

**Example of filter chaining**

You can also chain multiple filter blocks together to obtain a more specific set of data. The following image shows a chain of filters in the VPE.

For example, to filter out RFC1918 addresses (10.x.x.x, 172.16.x.x-172.31.x.x, and 192.168.x.x) and perform a **geolocate ip** action on the remaining addresses, perform the following steps:

1. From the **Start** block, create new filter block.
2. In the **Select Parameter** field, select **event > src_ip**.
3. Enter **10.0.0.0/8** in the **Select Value** field.
4. Create a second filter for the 172.16.0.0/12 IP addresses. This filter uses the filtered results from the previous block.
5. Create a third filter for the 192.168.0.0/12 IP addresses. This filter uses the filtered results from the previous block.
6. Create the **geolocate ip** action block on the remaining IP addresses.

If a filter block eliminates all variables while filtering, the downstream action can’t run.

**Example of using a custom list in a filter**

You can use custom lists in your **Filter** blocks to simplify checking against a fixed set of items. For example, instead of checking the source country of an IP address to see if it is North Korea, you can define a list of countries in a custom list, then check the IP address against all of the countries in the list. You can maintain custom lists using the Splunk Phantom web interface, REST API, and playbook API. See [Create custom lists for use in Splunk Phantom playbooks](#). The following image shows the VPE in Splunk Phantom.
In this example, use a custom list named Banned Countries in a filter by performing the following tasks:

1. Configure a geolocate ip action in the playbook editor.
2. Drag the half-circle icon on the geolocate ip action block in the editor.
3. Select Filter from the list of block types.
4. Click the Select Parameter field and select geolocate_ip_1.
5. Select geolocate_ip_1:action_result_data.*.country_name.
6. Leave == as the operator.
7. Click in the Select value field and select custom_list:Banned_Countries.

The country name of the source IP address is checked against the countries defined in the Banned Countries custom list.

**Use decisions to send artifacts to a specific downstream action**

Use a Decision block to change the flow of artifacts by performing IF, ELSE IF, or ELSE functions. When an artifact meets a True condition, it is passed downstream to the corresponding block in the playbook flow. If none of the Decision block conditions are met, the playbook run fails.

In this example, IP addresses found in artifacts are compared against some specific IP address ranges:

- If an IP address is in the 192.168.* range, perform an IP reputation lookup, ELSE IF
- The IP address is in the 172.16.* range, grant access to the user, ELSE
- If the IP address doesn't fall into either of the previous categories, perform a geolocate ip action.

The first time an artifact meets a condition, it is passed along to the corresponding downstream block. The artifact is no longer available for evaluation by other ELSE IF or ELSE statements, and cannot be passed to other downstream blocks. You can only perform one action on an artifact based on the condition that is matched first.

Unlike Filter blocks, no named datasets are created for reference later on in the playbook.

**Create a Decision block in your playbook**

To create Decision block, perform the following tasks:

1. Drag the half-circle icon attached to any existing block in the editor.
2. Select Decision from the list of block types.
3. Click the Select Parameter field and select the parameter you want to compare. Parameters are made available to the Decision block by upstream blocks.
4. Click the == field and select an operator for the decision.
5. Click the Select Value field and select the value you want to match. See Example of creating decisions for multiple downstream actions for an example of how these fields all work together.

6. (Optional) Click Add Else If to create another matching condition for the decision.

7. Click Add Else to create the final branch for the decision.

**Example of creating decisions for multiple downstream actions**

Decision blocks control the program flow based on comparisons of artifact data, notable properties, date functions, and action results. Create if and else if conditions to branch to multiple downstream blocks as a results of the comparisons.

In the following example, start with a Decision block that checks to see if any artifacts are in the notable. The image shows a decision block in the playbook editor.

1. Drag the half-circle icon attached to any existing block in the editor.
2. Select Decision from the list of block types.
3. In the If field, click the Select Parameter field and select a parameter to evaluate. You can choose from the properties provided by the container, event data, date and time options, and custom lists.
4. Select container properties from the list of options, and then click artifact_count as the property you want to evaluate.
5. Select > as the operator, and enter 0 in the Select Value field.

The blue circle next to the If section corresponds with the blue connector dot on the side of the decision block. All data is passed on to the next block.

**Example of creating decisions with multiple statements**

You can create more complex decision blocks with up to five statements. For example, you can perform a geolocate ip action on a source IP address and block the IP if the country is from North Korea. Otherwise, you can perform an ip reputation action on the IP address, as shown in the following screenshot:
Each subsequent statement and downstream block is color coded: blue marks the path of the If statement, and red marks the path of the Else statement. Each statement has its own and only one downstream block.

**Customize the format of your playbook content**

Use the **Format** block to craft custom strings and messages from various objects.

You might consider using a Format block to put together the body text for creating a ticket or sending an email. Imagine you have a playbook set to run on new containers and artifacts that does a basic lookup of source IP address artifacts. You want to take the results of that lookup, format the results, and send the information as an email. You craft your playbook so that the action results are available to the format block.

To configure a **Format** block, perform the following steps:

1. Create a new block in the Visual Playbook Editor (VPE).
2. Select **Format** from the list of block types.
3. Configure the template parameter variables in the **Template Parameters** field. The first variable is identified as `{0}`, the next as `{1}`, and so on. You can select any event and container properties. You can also select data from any upstream block.
4. In the **Template** field, craft a message using the variables you define.

**Example of defining a template**

This example defines a template in the following manner:

IP address: `{0}`
IP address country: `{1}`
IP address reputation: `{2}`

The message returned as a result of this template looks like the following:

IP address: 1.2.3.4
IP address country: United States
IP address reputation: Malicious

If multiple events are picked up at the same time, you see the following message:
IP address: 1.2.3.4, 10.11.12.13
IP address country: United States, Turkey
IP address reputation: Malicious

You can wrap %% around a formatting block to make each set of values output on its own line. For example:

%%
The IP address {0} originates from {1}.
%%

Wrapping %% around a formatting block produces results like the following:

The IP address 1.2.3.4 originates from United States.
The IP address 10.11.12.13 originates from Turkey.

**Example of using the Python str.format() function to create more advanced templates**

More complicated formatting is supported using all the capabilities of the Python str.format() function. The following template demonstrates double curly brackets to support curly bracket escaping, automatic escaping of backslashes and quotes, and centered string alignment with a custom filler character:

```
JSON Formatting:
{"notable_id": "{0}"}, "event_count": {1}}
```

Backslashes and quotes are escaped:
"ls /Applications/System \Preferences.app/*

Python formatting is supported:
```
|{0:-^50}|{1:-^50}|
|{2:-^50}|{3:-^{3}{2}{1}{3}{2}{1}50}|
```

The template produces output as in the following example:

```
JSON Formatting:
{"notable_id": 1004", "event_count": 10}
```

Backslashes and quotes are escaped:
"ls /Applications/System \Preferences.app/*

Python formatting is supported:
```
|-----------------------1004-----------------------|-----------------------10-----------------------|
|----------------------events----------------------|-------Zeus infection on HQ finance server------|
```

**Require user input to continue running the playbook**

You can configure the following types of user input in a playbook:

- A manual task using a **Manual Task** block that must be acknowledged by a user.
- A prompt using a **Prompt** block that must be acknowledged by a user. You can configure a specific response type with a **Prompt** block.
**Require user input using the Manual Task block in your playbook**

Use a Manual Task block to send a message to a Splunk Phantom user or group that they must acknowledge. This is the same as manually running a task action from the Investigation menu.

To configure a manual task, perform the following tasks:

1. Drop a new block onto the playbook editor.
2. Click on the block, then select Manual Task from the block types.
3. Select an Approver from the drop-down list. If the task is assigned to a group of users, the first user to process it completes the task.
4. From the Required response time field, choose the response time in minutes.
5. In the Message box, craft a meaningful message so the users receiving the message understand what actions they must take.

**Require user input using the Prompt block in your playbook**

Use a Prompt block in your playbook to send a message to a user or group that they must acknowledge.

To configure a prompt, perform the following tasks:

1. Drop a new block onto the playbook editor.
2. Click on the block, and then select Prompt from the block types.
3. Select an Approver from the drop-down list. If the task is assigned to a group of users, the first user to process it completes the task.
4. From the Required response time field, choose the response time in minutes.
5. In the Message box, craft a meaningful message so the users receiving the message understand what actions they must take.
6. From the Responses drop-down list, choose the type of response required to complete the task.

**Set container parameters using the API block**

Use the API block to set parameters of the container it's running in. For example, you can use an API call to set the severity of a container.

Perform the following tasks to configure an API block:

1. Drop a new block onto the playbook editor.
2. Click on the block, and then select API from the block types.
3. Select the API property you want to set. The following table summarizes the properties that you can set:

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>label</td>
<td>The label of the container. The drop-down list shows all of the container labels currently available on your Splunk Phantom instance.</td>
</tr>
<tr>
<td>sensitivity</td>
<td>The sensitivity of the container.</td>
</tr>
<tr>
<td>severity</td>
<td>The severity of the container.</td>
</tr>
<tr>
<td>status</td>
<td>The status of the container, such as Resolved.</td>
</tr>
<tr>
<td>owner</td>
<td>The owner of the container.</td>
</tr>
<tr>
<td>add list</td>
<td></td>
</tr>
<tr>
<td>Property</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>add list</td>
<td>One of two API calls that doesn't operate directly on the container. The <code>add list</code> property takes two parameters: the list that you want to add to, and the data you are adding. If the list doesn't exist, it is created by Splunk Phantom. You can point the data field to a variable by selecting from the properties, results, and artifacts, or you can type in a fixed string.</td>
</tr>
<tr>
<td>remove list</td>
<td>One of two API calls that doesn't operate directly on the container. The <code>remove list</code> property takes a list name as the single parameter, and deletes that list when it has run.</td>
</tr>
<tr>
<td>pin</td>
<td>Pin data to the heads-up display (HUD) in the container. This property takes the following parameters: Data, Message, Pin Type, Pin Style.</td>
</tr>
<tr>
<td>add tag</td>
<td>The API call used to add a tag to the container.</td>
</tr>
<tr>
<td>remove tag</td>
<td>The API call to remove a tag from the container.</td>
</tr>
<tr>
<td>add comment</td>
<td>The API call used to add a comment to a container. You can either supply a variable or a static string in the input.</td>
</tr>
<tr>
<td>promote to case</td>
<td>The API call used to promote the container to a case. It takes a single parameter, the case template you can pick from a drop-down list.</td>
</tr>
</tbody>
</table>

You can configure multiple API calls in any API block. For example, you can set the label, severity, and status of a container using one API block.

4. Click **Save** to save the settings. A check mark appears next to the API calls that you configured.

**Run other playbooks inside your playbook**

You can configure your playbook to run another existing playbook. Call one playbook from another playbook to avoid having to copy and maintain code in different places.

To configure your playbook to run another playbook:

1. Drop a new block onto the playbook editor.
2. Click on the block, then select **Playbook** from the block types.
3. In the **Playbook** field, select the playbook you want to run from the drop-down list.
4. (Optional) Toggle the **Synchronous** switch on to make this playbook wait for the called playbook to complete running before continuing.

Playbooks differ from action blocks in the following ways:

- The playbook continues to downstream blocks regardless of whether the called playbook is successful.
- The called playbook doesn't return any values that are used in downstream blocks.
- The called playbook doesn't determine the data set, and it operates on the container data with the scope inherited from the caller.
- The called playbook runs independently from the caller. If you wire a series of playbooks to run, they are processed in parallel if the **Synchronous** switch is left off. See **Determine your playbook flow**.

**Add custom code with the Custom Function block**

Add custom Python code to a **Custom Function** block. Custom functions enable you to expand the kinds of processing performed in a playbook, such as adding custom input parameters and output variables.
Functions, parameters, and output variables can all have custom names using the following characters:

- A-Z
- a-z
- 0-9
- Underscores

Setting or changing a custom name changes that custom name in all data paths that use it, including generated and custom code.

Set the name, tooltip, comments and other options for the block in the Advanced Settings > General Settings.

Click the $/>$ icon to open the Python Playbook Editor to add your custom code:

---

**Input parameters**

Input parameters represent a data path. You can set a data path from any valid blocks upstream, artifact data, and container data.

To create or remove an input parameter, perform the following steps:

1. Click the + icon to add an input parameter. The index of parameters starts at zero.
2. Type a name to set the name for or rename the input parameter.
3. Click the $>$ icon to set the properties for the input parameter.

You can delete the input parameter by clicking the x icon.

**Output variables**

Output variables are usable as inputs in other downstream blocks, such as Action, API, Filter, Decision, Format, Prompt, or other Custom Function blocks. The name of an output variable becomes `<block_name>__<variable_name>` in the auto-generated section of the playbook code. Give your output variables clear and meaningful names in your custom code.

Follow these steps to create or remove an output variable:

1. Click the + icon to add an output variable.
2. Type a name to set the name for, or rename the output variable.
3. Click the x icon to delete an output variable.

See the following example of a custom function:

```python
def format_login(action=None, success=None, container=None, results=None, handle=None, filtered_artifacts=None, filtered_results=None):
    phantom.debug('format_login() called')
    results_data_1 = phantom.collect2(container=container, datapath=['get_login_history:action_result.data.*.output'], action_results=results)
    results_item_1_0 = [item[0] for item in results_data_1]
    format_login__login_table = None

    # format the output into Jira's markup language for rendering a table
    format_login__login_table = "|| output of '/usr/bin/last -a' ||\n"
    last_lines = results_item_1_0[0].split('\n')
    for line in last_lines:
        format_login__login_table += "| {} |\n".format(line)
    phantom.debug("table of logins for jira:")
    phantom.debug(format_login__login_table)

    phantom.save_run_data(key='format_login:login_table', value=json.dumps(format_login__login_table))
    enrich_ticket_1(container=container)
    return
```

Determine your playbook flow

The order in which you arrange the blocks and lines in your playbook determine the playbook flow.

Process playbook blocks serially

Serial processing means playbook blocks are performed in the order they are arranged, as shown in the following screenshot:

In this example, the blocks perform as described:

1. A geolocate ip action is performed on a source IP address.
2. When the geolocate ip action is finished, a lookup ip performs.
Use serial processing when the operations must happen in a specific order, such as when a downstream block depends on the results from an upstream block.

**Processing playbook blocks in parallel**

You can also wire blocks to process in parallel, as shown in the following example:

![Playbook diagram](image)

In this case, the `geolocate ip` and `lookup ip` actions perform simultaneously, and either action can finish first. You can wire blocks in this manner when you have no dependencies on the completion of either block, or if there are no dependencies between the blocks themselves.

**Arranging blocks in a playbook**

You can drag blocks around the canvas. Lines connected to boxes automatically arrange themselves when you move blocks around.

Hover over any playbook block and click the trash can icon to delete the block. The corresponding connecting arrow is also deleted.

**Save a playbook so that Splunk Phantom can access it**

You must save a playbook before Splunk Phantom can use it. If the playbook has errors, an error message appears with additional information. You cannot save a playbook until you resolve all errors.

1. Click **Save** to begin saving a playbook.
2. Enter a brief description of the playbook.
3. (Optional) Click **Reuse this comment until the editor is reloaded** if you want to use the same description each time you save the playbook.
4. Click **Save** to finish saving the playbook.

**Create a copy of an existing playbook**

You can create a copy of any saved playbook by using the **Save As** option.

Perform the following tasks to create a copy of an existing playbook:

1. Click the down arrow next to the **Save** button and click **Save As**.
2. Enter a new name for the playbook.
3. Select the repository where you want to save the playbook.
4. (Optional) Enter comments or a description about the playbook.
5. Click **Save**.

**View or edit playbook settings**

To view or edit playbook settings after you've saved a playbook, click **Playbook Settings**. You can also view playbook settings before a playbook is saved, but not all fields are available until after the playbook is saved.

The following table describes the fields in the playbook settings.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operates on</td>
<td>Related information in Splunk Phantom is organized in containers. Playbooks contain the list of artifacts the playbook work on and the results of the playbook and action runs. A playbook can't run without an associated container, which holds the inputs and outputs for a playbook run. Containers also have a label associated with them, which is used to group together different kinds of information. For example, Splunk Phantom includes one default notable label, <strong>Events</strong>. Other labels could be <strong>Intelligence</strong> for data from threat and intel feeds or <strong>Phishing</strong> for phishing emails. Playbooks are designated to run on particular labels. Select which labels this playbook works on from the <strong>Operates on</strong> field. Most playbooks are designed to work on a particular category, and therefore a particular label.</td>
</tr>
<tr>
<td>Tenants</td>
<td>Select one or more tenants to run the playbook against the containers belonging to the selected tenants. Use an asterisk (*) to run the playbook on containers for all tenants. See Configure multiple tenants on your Splunk Phantom instance in Administer Splunk Phantom for more information about configuring multiple tenants.</td>
</tr>
<tr>
<td>Category</td>
<td>Use categories to organize and save your playbooks into folders. For example, you can create a <strong>Production</strong> category for playbooks that are ready to be marked active, and a <strong>Test</strong> category for playbooks that are under development.</td>
</tr>
<tr>
<td>Run as</td>
<td>The service account used by Splunk Phantom to run the playbook.</td>
</tr>
<tr>
<td>Logging</td>
<td>Toggle this switch to turn on debug logging each time the playbook is run. Logging might be useful when you create a new playbook. Later, you can turn logging off to save disk space.</td>
</tr>
<tr>
<td>Active</td>
<td>The playbook will automatically run on every new container or artifact that comes into Splunk Phantom, for the playbook labels and tenants it is set to run on.</td>
</tr>
<tr>
<td>Safe Mode</td>
<td>Toggle this switch to put the playbook in read-only mode. By turning on Safe Mode, the playbook will be unable to run read-write actions. Read and write actions are defined by each app in Splunk Phantom. For example, in an LDAP app, <code>get users</code> is a read-only action, while <code>reset password</code> is read/write.</td>
</tr>
<tr>
<td>Draft Mode</td>
<td>Toggle this switch to save a draft of your playbook, even if your playbook is incomplete or has errors. Playbooks in draft mode can't be marked active.</td>
</tr>
<tr>
<td>Description</td>
<td>Enter a description for the playbook. The description becomes a triple-quoted comment in the playbook and appears on the playbooks page.</td>
</tr>
<tr>
<td>Notes</td>
<td>Notes can be viewed only by editing the playbook.</td>
</tr>
<tr>
<td>Export Playbook</td>
<td>You can share playbooks by exporting them. Import a shared playbook file on the playbooks page.</td>
</tr>
<tr>
<td>Revision History</td>
<td>Click <strong>View</strong> to see a previous revision of the playbook. You can make edits and save as a new version, or click <strong>Latest Version</strong> to return to the most current version. Click <strong>Revert</strong> to use the corresponding previous version of the playbook as the most current version.</td>
</tr>
<tr>
<td>Audit Trail</td>
<td>The Audit Trail button downloads a CSV file that shows the full audit trail of the playbook, including dates and times.</td>
</tr>
<tr>
<td>Docs</td>
<td>Click the <strong>Docs</strong> link to go to the documentation page for Splunk Phantom.</td>
</tr>
</tbody>
</table>
View or edit the Python code in Splunk Phantom playbooks

Click the Python Playbook Editor tab to view the underlying Python code for your playbook. The code for the entire playbook is shown by default. Click any block in your playbook to view the code for the selected block only.

The on_start function is similar to the main or begin functions in other programming languages. Splunk Phantom executes the Python code in your playbooks by calling the on_start function with the container context.

Manage your editing session

Use the icons in the Python Playbook Editor to manage your editing session.

<table>
<thead>
<tr>
<th>Icon</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="View the Python code for the entire playbook. Using this icon is useful if you are viewing the Python code for a specific block on the canvas, and want to return to view the Python code for the entire playbook." /></td>
<td>View the Python code for the entire playbook. Using this icon is useful if you are viewing the Python code for a specific block on the canvas, and want to return to view the Python code for the entire playbook.</td>
</tr>
<tr>
<td><img src="image" alt="Add code that needs to be defined at the global level of the playbook, such as import statements for Python libraries." /></td>
<td>Add code that needs to be defined at the global level of the playbook, such as import statements for Python libraries.</td>
</tr>
<tr>
<td><img src="image" alt="View functions for blocks that have diverging or converging actions. The functions are explained in the following list:" /></td>
<td>View functions for blocks that have diverging or converging actions. The functions are explained in the following list:</td>
</tr>
<tr>
<td>- <strong>Block Function</strong> is highlighted when viewing the Python code that is applicable to a single block.</td>
<td></td>
</tr>
<tr>
<td>- <strong>Callback Function</strong> is used to view to the block of code that is generated to split the output of the single block into multiple blocks.</td>
<td></td>
</tr>
<tr>
<td>- <strong>Join Function</strong> is used to view the block of code that is generated to join the output of the multiple blocks into a single block.</td>
<td></td>
</tr>
<tr>
<td><img src="image" alt="Go back to the original version and discard all changes. If there are changes to revert, the button turns white when you hover over it." /></td>
<td>Go back to the original version and discard all changes. If there are changes to revert, the button turns white when you hover over it.</td>
</tr>
</tbody>
</table>

How custom Python edits affect the visual playbook editor

When you see Full Code in the Python Playbook Editor, you are making changes affecting the whole playbook. When you begin to make edits, you are prompted to verify that you want to continue, which disables the visual playbook editor (VPE) for the playbook. Now, you can only edit the Python code.

If you click a block in the playbook, your edits only disable the VPE for that block. The Python Playbook Editor changes from Full Code to the name of the Python function called in that block. You can continue to use the VPE to add, edit, or delete other blocks. If you want to add another block downstream from the block you edited, you have to manually enter a Python function call for the next block, such as phantom.act(). The VPE doesn't generate Python code for any block containing custom edits.

When editing the Python code for a Custom Function block, make your edits in the editable area in order for callback functions to work.

1. Create a Custom Function block in the VPE. See Add custom code with the custom function block.
2. Click Python Playbook Editor.
3. Click the Custom Function block.
4. Write your custom code in the area with the # Write your custom code here... text.

```python
# Custom Code Start
# Write your custom code here...
# Custom Code End
```

33
Manage cases in Splunk Phantom

Overview of cases

Containers can be promoted to cases. You can use cases to consolidate information from multiple containers. Cases have phases and tasks, which are organized into workbooks to track and manage all the actions taken. Tasks can have playbooks and actions associated with them, allowing you to automate these actions. Automating actions allows Splunk Phantom to be used to track policy and compliance, and to fulfill documentation requirements.

Create cases in Splunk Phantom

Once you have at least one case workbook, you can create cases to use that workbook.

Cases only contain the items from the workbook at the time the case was created. If you create a case from a workbook, and then later add a new phase to the workbook, the new phase is not available to the existing workbook. Only new cases created after the workbook is changed will have the new phase available to use. The case was a copy at the time it was created. There is no live link to the workbook. Items deleted from the workbook aren’t deleted from cases created before the workbook change.

Promote a container to a case

Create a case by promoting a container.

1. From the main menu, select Sources, and then select a container label.
2. Click the suitcase ( ) icon.
3. In the Promote to Case window, select the new workbook you want to use on this case. If you already added a workbook to the container, you do not have the option to select a workbook. The menu is inactive with the text "Keep current workbook".
4. Click Save.

A case looks similar to its container and has all of the same functions. The colored block with the word Case indicates that it is a case.

Select the Workbook tab to see the tasks defined in case workbook. The blue highlight indicates the current page and shows task completion progress within each phase.

Demote a case to change it back to a container

Perform the following steps to change a case back to a container:

1. In Splunk Phantom, navigate to the case you want to demote.
2. Click the suitcase ( ) icon.

Delete a case in Splunk Phantom

Perform the following steps to delete a case:
1. In the main menu, select **Cases**.
2. Select the cases you want to delete.
3. Click **Delete**.
4. Click **Delete** again to confirm that you want to delete the selected cases.

**Add objects to a case in Splunk Phantom**

Add objects to a case in one of the following ways:

- Promote a container to a new case. Everything in the container becomes a case object.
- Promote a container to an existing case. Choose the objects from the container to be copied to the existing case. The container itself remains a container and is not promoted to a case.
- Copy an individual object to an existing case with the Add to Case option.

**Add objects from a container to an existing case**

Perform the following steps to add objects from a container to an existing case:

1. Navigate to a container in Splunk Phantom.
2. Click the suitcase icon.
3. Select the case in the **Add Event to Case** dialog box:
   1. Select **Existing Case**.
   2. In the **Case Name** field, select an existing case, or start typing to filter the case names before selecting a case.
   3. Select a phase from the case that you want to add objects to.
   4. Select the object type from the container that you want to add to the case. If the object is evidence, check the **Mark as evidence** checkbox.
4. Click **Save**.

You can add objects from a container to a case only once. If you try to add objects from the same container to the same case, an error message appears.

See [Create cases in Splunk Phantom](#) for information about promoting an entire container to a case.

**Add artifacts from a container to a case**

Perform the following steps to add artifacts from a container to a case:

1. Navigate to a container in Splunk Phantom.
2. Click **Analyst** to change the container to the analyst view.
3. Click the **Artifacts** tab.
4. Click the ... icon on the artifact line, and then select **Add To Case**.
5. Complete the **Add Artifact to Case** dialog box:
   1. Click the **Case Name** field and select an existing case, or start typing to filter the case names before selecting a case.
   2. Select a phase from the case that you want to add artifacts to.
   3. (Optional) Click **Include note** and add a note to accompany the artifact being added.
   4. (Optional) If the artifact is evidence, check the **Mark as evidence** checkbox.
6. Click **Save**.

36
Add files from a container to a case

Perform the following steps to add files from a container to a case:

1. Navigate to a container in Splunk Phantom.
2. Click Analyst to change the container to analyst view.
3. Click the Files tab.
4. Click the ... icon on the artifact line, and then select Add To Case.
5. Complete the Add File to Case dialog box:
   1. Click the Case Name field and select an existing case, or start typing to filter the case names before selecting a case.
   2. Select a phase from the case that you want to add the file to.
6. Click Save.

Add action results from a container to a case

Perform the following steps to add action results from a container to a case:

1. Navigate to a container in Splunk Phantom.
2. Click Analyst to change the container to analyst view.
3. Click the Activity tab. Action run results appear near the bottom in the Activity tab.
4. Click the ... icon on an action result and select Add To Case.
5. Complete the Add Action Result to Case dialog box:
   1. Click the Case Name field and select an existing case, or start typing to filter the case names before selecting a case.
   2. Select a phase from the case that you want to add the file to.
6. Click Save.

Define a workflow in a case using workbooks in Splunk Phantom

You can define a workflow in a case by using workbooks. Workbooks are lists of standard tasks that you follow when you evaluate events or cases. You can create workbooks to analyze events. You can also combine multiple workbooks to create a more comprehensive workbook for cumulative events, cumulative cases, or cases that start out as one type of incident but end up as a different type of incident.

Workbooks are available from Investigation in both Summary View and Analyst View.

Add a workbook to an event or case

Perform the following steps to add a workbook to an event or case:

1. Navigate to an event or case in Splunk Phantom.
2. Click the Workbook tab.
3. Click Add Workbook.
4. Select the desired workbook from the drop-down list.
5. Click Save.

In Analyst View in Investigate, you can click Add to add additional workbooks to the event, or click Edit to make changes to the workbook. If you edit a workbook in this manner, the changes only apply to the current event, not for all events. You must edit a workbook on the Workbooks page to make global changes. See Define tasks using workbooks in Administrator
Splunk Phantom.

**Use workbooks to track, edit, and complete tasks**

Use workbooks in a case to track, edit, and complete tasks after you have added items to the case.

Perform the following tasks to view the workbook for a case:

1. Navigate to the case in Splunk Phantom.
2. Click Analyst to switch to the Analyst View.
3. Select the Workbook tab.

**Add new workbooks or edit phases and tasks**

You can add existing workbooks to a case, add new phases to a workbook, or manage tasks.

1. Navigate to the case in Splunk Phantom.
2. Click Analyst to switch to Analyst View.
3. Select the Workbook tab.
4. Click Add to add existing workbooks to the case.

If you have created self-contained workbooks to analyze certain types of incidents, adding multiple workbooks is useful for cases that start out like one type of incident but turn out to be a different type of incident. This helps you avoid any inconsistencies that might occur from adding individual phases or tasks during analysis. It is also possible to add individual phases or tasks.

Click Edit to add new phases or manage tasks. You can add, remove, or rename tasks, assign an owner to a task, assign authorized users, or configure whether or not a note is required for the task to be completed. If you edit a workbook in this manner, the changes only apply to the current event, not for all events. You must edit a workbook on the Workbooks page to make global changes.

**Manage task details**

Click on a task in the workbook column to open the task details in the main window area. You can view the task name and description supplied when the task was created.

1. Navigate to the case in Splunk Phantom.
2. Click Analyst to switch to Analyst View.
3. Select the Workbook tab.
4. Click the name of the task.
5. Select a progress status from the drop-down list.

All tasks start with the status of Incomplete by default. As you complete tasks, additional options such as In-Progress or Complete become available. If configured to do so, some items require you to enter a note before you can mark it as complete.

A checkmark next to the task name indicates that it is complete. You can change the status of a task to Incomplete if the task requires additional information or action.
Create case reports to download and share in Splunk Phantom

Create a case report as a PDF file that can be downloaded and shared. You can generate a case report at any time from either Summary or Analyst view. The report contains all of the information in the case up to that point. You can have multiple reports per case.

Perform the following tasks to generate a case report:

1. Navigate to the case in Splunk Phantom.
2. Click + Report.
3. (Optional) Click Include data to generate an archive containing the PDF report file and copies of all the data in the case, such as files, action run results, copies of artifacts in JSON format, and notes. Including data in a case report can significantly increase the size of the report and the time it takes to generate the report.
4. Click Generate.
Use Splunk Phantom in a Connected Experiences App

Use the Splunk Mobile app for Splunk Phantom

Use the Splunk Mobile App to view and respond to notifications, view dashboards, view event details, or run a playbook in Splunk Phantom.

Prerequisites

Before using the Splunk Mobile app for Splunk Phantom, perform the required administrative tasks. See About the Splunk Mobile App for Splunk Phantom in the Get Started with the Splunk Mobile App for Splunk Phantom manual.

View a notification

View a notification by opening a push notification in the Splunk Mobile app. Or, you can open a notification in the Splunk Mobile UI.

1. In your Splunk Phantom instance in the Splunk Mobile app, navigate to the Notifications tab. You can filter notifications by type by tapping All Types at the top of the list.
2. Tap a notification to view its details.

Respond to a notification

You can also respond to notifications in the Splunk Mobile app.

1. In your Splunk Phantom instance in the Splunk Mobile app, navigate to the Notifications tab.
2. Select a notification.
3. To respond to the notification, complete the fields that the notification requests.

View dashboards

To view your Splunk Phantom dashboards, navigate to the Dashboards tab in the Splunk Mobile app.

Depending on the visualization type, you can scroll through or tap the visualization to get more details.

View event details

You can view event details from Splunk Phantom on your mobile device using the Splunk Mobile app.

To view an event, perform these steps:

1. In your Splunk Phantom instance in the Splunk Mobile app, navigate to the Events tab. You can filter events by owner and status at the top of the list.
2. Tap an event to view its information.

To run a playbook against the event, tap the Playbook button. See Run a playbook for more information about running playbooks on your mobile device.
Tap the **Activity** tab to view event activities or add a comment. Tap the **Artifacts** tab to view event artifacts. Tap the **Notes** tab to view and filter event notes.

**Run a playbook**

You can run a playbook that you create in Splunk Phantom on your mobile device using the Splunk Mobile app.

Follow these steps to run a playbook in the Splunk Mobile app:

1. Create playbooks in Splunk Phantom. See [Use the Visual Playbook Editor to create and debug playbooks](#).
2. In your Splunk Phantom instance in the Splunk Mobile app, navigate to the **Events** tab. You can filter events by owner and status at the top of the list.
3. Tap an event that you want to run a playbook in response to.
4. Tap the **Playbook** button.
5. Select the playbook you want to run.
6. Select the scope of the playbook. The scope indicates which artifacts the playbook processes. New includes only artifacts from when this playbook was last run. All includes all artifacts. Artifact processes a specific artifact defined by the artifact ID.
7. Tap **Run Playbook**.

The Splunk Mobile app runs the playbook against the event you selected.

**Run Splunk Phantom playbooks in Splunk AR workspaces**

Workflow automation is a beta feature available in Splunk AR version 2.1.0. Workflow automation integrates Phantom playbooks into AR workspaces to guide users through real-world tasks. To use workflow automation, create playbooks in Phantom and then add them to your AR workspaces in the Splunk AR mobile app.

See [Add Phantom playbooks to AR workspaces in Splunk AR (beta)](#) to learn how to add playbooks to AR workspaces.

See [Run a Splunk Phantom playbook in Splunk AR](#) for instructions on how to run a playbook in Splunk AR.
Use the command line interface to perform tasks in Splunk Phantom

Splunk Phantom command-line interface overview

Analysts can perform a number of tasks from either the command line of the *nix shell or from the comments field of a container through the PhBot CLI interpreter in Splunk Phantom.

The command-line interface in Splunk Phantom supports a number of tasks:

- Run an action
- Run a playbook
- Add a note to a container
- Update or edit a container
- Get datapath information for use with other actions

Each task type has an associated slash command and arguments.

Use the CLI tool in Splunk Phantom

You can access the command line interface from the Linux shell by running a script with the required command and arguments. You can find the script in `<PHANTOM_HOME>/bin/run_slash_command.pyc`.

If you choose to use the CLI tool from a Linux shell, you are prompted to authenticate for each command unless the PH_AUTH_TOKEN or PHANTOM_USERNAME and PHANTOM_PASSWORD environment variables are set. Use the username and password for a valid Phantom user, which might be different from the Linux user account. User credentials are not cached.

You can set environment variables for the Linux user account using the `setenv` command or by editing the user account’s profile.

You can also use the PH_AUTH_TOKEN environment variable for a temporary session as shown in the following example:

```
export PH_AUTH_TOKEN="<token>"
```

```
phenv python run_slash_command.pyc --help
```

If you use sudo to use slash commands and want to use the PH_AUTH_TOKEN environment variable, use the `-E` argument to preserve the environment variable.

Anatomy of a slash command

A slash command is an instruction that begins with a forward slash (`/`) followed by a predefined command then any required or optional arguments. Each command has a series of arguments needed for the execution of the command. The order of arguments is important.

You can use the `--help` argument with a slash command to determine which arguments are needed and in what order they need to be listed.
Use the following format for the action slash command:

```
/action < action_name > < app > < required arguments > < --asset asset_name> < --optional arguments >
```

This example shows the slash command `/action` followed by the `action_name`, then the required app to run the action, and finally the required arguments.

```
/action geolocate_ip "MaxMind" 1.1.1.1
```

**Slash command examples with the CLI tool**

Run the `run_slash_command.pyc` script without arguments to get the help output.

**Command:**

```
phenv python run_slash_command.pyc
```

**Output:**

```
run_slash_command.pyc USAGE: <container-id> <slash-command>

You will be prompted for authentication. You can set the following environment variables to avoid this:

Environment:
PH_AUTH_TOKEN:    Authenticate using an auth token.
PHANTOM_USERNAME: Authenticate with user name. Requires PHANTOM_PASSWORD set to avoid prompt.
PHANTOM_PASSWORD: Authenticate with password.

Hint: You can get the container ID from the phantom event UI, /mission/<container-id>/

Examples:
- phenv run_slash_command.pyc 1 /action geolocate_ip "MaxMind" 1.1.1.1
- phenv run_slash_command.pyc 1 /playbook 12 all
- phenv run_slash_command.pyc 1 /set name "My Container Name"
- phenv run_slash_command.pyc 1 /note "Errant IPs" IPs encountered include 'artifact:*.network.src_ip'
- phenv run_slash_command.pyc 1 /inspect 'artifact:*'
- phenv run_slash_command.pyc 1 /inspect '[1, 2, 3, 4, 5]'
```

Use the CLI tool to add a note to a container.

**Command:**

```
phenv python run_slash_command.pyc <container ID> /note "Errant IPs" IPs encountered include 'artifact:*.network.src_ip'
```

**Output:**

```
[2019-12-12 00:02:08] Execution result was:

Command finished successfully!
```

The Splunk Phantom web interface has a new note for the container with the title "Errant IPs" and the body:
Use the CLI in the Splunk Phantom web interface

Analysts can use the CLI from the comments field on events or cases in Investigation. When using this method to run commands, slash commands run against the current container.

In the web-based interface, slash commands support auto-completion of options and arguments. Results are displayed in the activity sidebar.

Run an action in Splunk Phantom

Analysts can use the /action command to quickly run one of the actions Splunk Phantom supports.

Actions run with /action are the same actions that are found in the Run Action dialog box, but the names of the actions are formatted with underscores ( _ ) instead of spaces. For example, the action geolocate ip becomes geolocate_ip.

The Run Action dialog box guides you through selecting the information an action requires. Using the command line interface requires you to provide the same information as arguments to the /action command.

When you type /action in the comment field of the activity sidebar, a tooltip-style dialog appears to guide you through adding arguments, or you can use the --help argument to get a message with help information as shown here:

```
$action geolocate_ip "MaxMind" --help
```

PhBot returns the following help message:

```
usage: /action geolocate_ip [app] <required arguments> [--asset asset...] [--optional arguments]
Queries MaxMind for IP location info

required arguments:
ip IP to geolocate

The command-line interpreter validates arguments with the /action command. Incorrect arguments generate an error message to help you fix the arguments as shown in the following example:

```
$action whois_domain "WHOIS" splunk.com
The following error message is returned for the example:

$action whois_ip "WHOIS" a.b.not_an_ip
```

Use a list with the /action command

You can perform actions on lists of items by passing the list as an argument as shown in the following example:

```
$action geolocate_ip "MaxMind" ["1.1.1.1", "2.2.2.2"]
```

Lists must be presented in valid Python syntax, so individual items must be in quotation marks ( " ).

Passing the /action command multiple lists or datapaths, or a mix of lists and datapaths, results in a product. For example, [1, 2] [3, 4] results in four action runs: (1, 3), (1, 4), (2, 3), and (2, 4).
Run a playbook in Splunk Phantom

Analysts can use the `/playbook` command to run a playbook from the command line in Splunk Phantom.

To run a playbook from the command line, you must supply the playbook_id or playbook_name and the scope. A playbook_name consists of a repository, followed by a slash (/), and the name of the playbook.

You can get a playbook_id or playbook_name by looking up the playbook from Main Menu > Playbooks, and clicking the playbook name from the list. The ID is the number in the playbook URL. See the following example:

https://<phantom.example.com/playbook/1

Or you can use the REST API to query `/rest/playbook`. See Query for Data in Splunk Phantom REST API Reference.

Scope is one of the following values:

- **new** - Run the playbook for only artifacts added to the container since the last time the playbook was run.
- **all** - Run the playbook against all artifacts in the container.
- `<artifact ID>` - Run the playbook for either a specific artifact or a list of artifacts.

**Example using the playbook ID**

`/playbook 1 new`

**Example using the playbook name**

`/playbook local/example_playbook all`

You can also supply lists for IDs or scope to run multiple playbooks, to run a playbook for multiple specified artifacts or scopes, or multiple playbooks for multiple specified artifacts.

**Example of multiple specified artifacts**

`/playbook 1 ["41", "43", "45"]`

This example runs playbook 1, for artifact IDs 41, 43, and 45 in the container.

**Example of multiple playbooks**

`/playbook ["1", "2", "3"] new`

This example runs playbooks 1, 2, and 3 for new artifacts in the container.

**Example of multiple playbooks and multiple scopes**

`/playbook ["1", "2"] ["new", "all"]`

The example runs playbooks 1 and 2 for both the new and all scope.

Add a note in Splunk Phantom

Add a general note using the `/note` command in Splunk Phantom. Only general notes are supported. Use the following format:

`/note "<title>" <note body>`
You can use a datapath with a note to add additional information to a note. See Use a datapath in Splunk Phantom. This is shown in the following example:

/note "Attackers" Based on geolocate ip, attacks originated from artifact:*.ip
The above example results in a note added with the title "Attackers" and a body that looks like the following:

Based on geolocate ip, attacks originated from [2.2.2.2, 1.1.1.1]

Notes and datapaths

You can use a datapath anywhere in a note title or body. The datapath is evaluated as a Python style list, and creates a single note with the results listed in it.

See Use a datapath in Splunk Phantom.

Update or edit an event in Splunk Phantom

You can edit or set several attributes of an event, also called a container, using the /set command.

You can set or edit these attributes:

- name
- label
- owner_id
- status
- severity
- sensitivity

Use the following format to set an attribute:

/set <attribute> <value>

You can use datapaths to set attributes for multiple events at a time. See Use a datapath in Splunk Phantom.

Examples

Rename a container

/set <current name> <new name>

Set the severity of an event

/set severity high

Set the status of an event

/set status open

Use a datapath in Splunk Phantom

You can use a datapath as an argument with a slash command. This makes slash commands flexible and powerful. Use the /inspect command to get the datapath to use with other slash commands.
**Datapaths**

A datapath is a series of names, keywords, attributes, and wildcards that evaluates to a list of values. These values can be attributes of artifacts or action results.

A datapath is described using this format:

\(<type>:<path.to.value_name>\)

**Example of a datapath for an artifact attribute:**

artifact:*.ip

Artifacts are indexed by their common event format (CEF) data so only CEF data is available to use in datapaths. You cannot access other fields such as label or description.

**Example of a datapath for action results:**

action_result:data.*.longitude

For more information on datapaths, see collect in the *Splunk Phantom Playbook API Reference.*

**Use the /inspect command**

Use the /inspect command to examine artifacts and to look for datapaths you want to use with another slash command.

See the following example:

Examine an artifact to see if it has IP addresses in its CEF data.

```
/inspect artifact:*
```

The above example returns the following:

```
{u'ip': u'2.2.2.2'}
{u'ip': u'1.1.1.1'}
```

Because there is IP information in the artifact, you can access that information in another command with a datapath.

```
/action whois_ip "WHOIS" artifact:*.ips.*
```

See the following example:

Examine an action_run.

```
/inspect action_run:1
```

**Returns:**

JSON formatted action run information.

```
{
    "comment": "",
    "node_guid": "d7c64d0f-fd0b-4d0b-8c68-34704ee91247",
    "playbook_run": null,
    "exec_order": null,
}
See the following example:

Get a list of all app_runs.

/inspect app_run:*.id

Returns:

4
7
6
You can use these app_run IDs with other commands or REST API calls.