Kelverion

INTEGRATION PACK FOR CISCO PRIME INFRASTRUCTURE

For Microsoft System Center Orchestrator

For System Center 2016 and 2019, you must use the 32-bit version of the integration pack, which has the name **Kelverion_Integration_Pack_for_Cisco_Prime_Infrastructure_2.1**

For System Center 2022 and later, you must use the 64-bit version of the integration pack, which has the name **Kelverion_IP_Cisco_Prime_Infrastructure_x64_2.1**

User Guide

Version 2.1

Kelverion Integration Pack for Cisco Prime Infrastructure

Published: June 2024

Feedback

Send suggestions and comments about this document to support@kelverion.com.

Contents

Kelverion Integration Pack for Cisco Prime Infrastructure

The Integration Pack for Cisco Prime Infrastructure is an add-on for System Center Orchestrator that enables you to integrate with Cisco Prime Infrastructure.

System Requirements

The Integration Pack for Cisco Prime Infrastructure requires the following software to be installed and configured prior to implementing the integration. For more information about installing and configuring Orchestrator and Cisco Prime Infrastructure, refer to the respective product documentation.

Kelverion_Integration_Pack_for_Cisco_Prime_Infrastructure (32-bit)

- Microsoft System Center Orchestrator 2016, 2019
- Microsoft .NET Framework 4.7.2
- Cisco Prime Infrastructure 3.8, 3.9, 3.10

Kelverion_IP_Cisco_Prime_Infrastructure_x64 (64-bit)

- Microsoft System Center Orchestrator 2022
- Microsoft .NET Framework 4.7.2
- Cisco Prime Infrastructure 3.8, 3.9, 3.10

Registering and Deploying the Integration Pack

After you download the integration pack file, you must register it with the Orchestrator management server and then deploy it to Runbook Servers and Runbook Designers. For more information about how to install integration packs, see the How to Install an Integration Pack in the online documentation for System Center Orchestrator.

IMPORTANT: Ensure that you are deploying the correct version of the Integration Pack.

- For System Center 2016 and 2019, you must use the 32-bit version of the integration pack, which has the name **Kelverion_Integration_Pack_for_Cisco_Prime_Infrastructure**
- For System Center 2022 and later, you must use the 64-bit version of the integration pack, which has the name **Kelverion_IP_Cisco_Prime_Infrastructure_x64**

To register the integration pack:

1. On the management server, copy the **.OIP** file for the integration pack to a local hard drive or network share.
2. Confirm that the file is not set to **Read Only** to prevent unregistering the integration pack later.
3. Start the **Deployment Manager**.

4. In the navigation pane of the Deployment Manager, expand **Orchestrator Management Server**, right-click **Integration Packs** to select **Register IP with the Orchestrator Management Server**. The **Integration Pack Registration Wizard** opens.
5. Click **Next**.
6. In the **Select Integration Packs or Hotfixes** dialog box, click **Add**.
7. Locate the **.OIP** file that you copied locally from step 1, click **Open** and then click **Next**.
8. In the **Completing the Integration Pack Wizard** dialog box, click **Finish**.
9. On the **End User Agreement** dialog box, read the Kelverion License Terms**,** and then click **Accept**.
10. The **Log Entries** pane displays a confirmation message when the integration pack is successfully registered.

To deploy the integration pack:

1. In the navigation pane of the **Deployment Manager**, right-click **Integration Packs**, click **Deploy IP to Runbook Server or Runbook Designer**.
2. Select the integration pack that you want to deploy, and then click **Next**.
3. Enter the name of the runbook server or computers with the Runbook Designer installed, on which you want to deploy the integration pack, click **Add**, and then click **Next**.
4. Continue to add additional runbook servers and computers running the Runbook Designer, on which you want to deploy the integration pack. Click **Next**.
5. In the **Installation Options** dialog box, configure the following settings.
6. To choose a time to deploy the integration pack, select the **Schedule installation** check box, and then select the time and date from the **Perform installation** list.
7. Click one of the following:
    a. **Stop all running runbooks before installing the integration pack** to stop all running runbooks before deploying the integration pack.
    b. **Install the Integration Packs without stopping the running Runbooks** to install the integration pack without stopping any running runbooks.
8. Click **Next**.
9. In the **Completing Integration Pack Deployment Wizard** dialog box, Click **Finish**.
10. When the integration pack is deployed, the **Log Entries** pane displays a confirmation message.

Licensing the Integration Pack

After you register and deploy the integration pack you must provide a valid Kelverion license before running any runbooks that contain activities from the integration pack.

To deploy the integration pack license file to System Center Orchestrator 2019 or earlier:

1. Copy the .KAL license file to %PROGRAMFILES(X86)%\Kelverion Automation\Licenses
2. Repeat for each Orchestrator Runbook Server and Runbook Designer host system.

To deploy the integration pack license file to System Center Orchestrator 2022 or later:

1. Copy the .KAL license file to %PROGRAMFILES%\Kelverion Automation\Licenses
2. Repeat for each Orchestrator Runbook Server and Runbook Designer host system.

Configuring the Integration Pack

A connection establishes a reusable link between Orchestrator and a Cisco Prime Infrastructure server. You can create as many connections as you require specifying links to multiple servers. You can also create multiple connections to the same server to allow for differences in security permissions for different user accounts.

To set up a Cisco Prime Infrastructure configuration:

1. In the Client, click the **Options** menu, and select *KA Cisco Prime Infrastructure*. The **KA Cisco Prime Infrastructure** dialog box appears.
2. On the **Configurations** tab, click **Add** to begin the configuration setup. The **Add Configuration** dialog box appears.
3. In the **Name** box, enter a name for the configuration. This could be the name of the Spectrum server or a descriptive name to distinguish the type of configuration.
4. Click the ellipsis button (…) next to the **Type** box and select *Cisco Prime Infrastructure Configuration.*
5. In the **Configuration File Path** box enter the file path to the IP configuration XML file. For more information see Additional Configuration.
6. In the **Cisco Prime Infrastructure Server** box, type the name or IP address of the Prime Infrastructure server.
7. In the **User Name** and **Password** boxes, type the credentials that Orchestrator will use to connect to the Cisco Prime Infrastructure server. The user should be configured in Cisco Prime Infrastructure to be part of the following groups: Admin, Config Managers, NBI Credential, NBI Read, NBI Write.
8. In the **Request Timeout (seconds)** box, enter the number of seconds the IP should wait for a request, before timing out.
9. In the **Max Batch Results** box, enter the maximum number of records that should be returned in one request (batch). Minimum allowed value is 50. Maximum allowed value is 1000. When retrieving a large set of records, if the number of records to be retrieved is greater than the value specified Max Query Results, multiple requests will be executed to retrieve the entire record set.
10. In the **Batch Interval (milliseconds)** box, enter the number of milliseconds to wait between multiple batch requests. For details see Rate Limiting Restrictions.
11. In the **Max Retry Count** box, enter the maximum number of times the IP should retry the request when encountering a rate limiting error (*503 Service Unavailable)*. For details see Rate Limiting Restrictions.

12. In the **Retry Interval (seconds)** box, enter the number of seconds the IP should wait between retries, when encountering a rate limiting error (*503 Service Unavailable)*. For details see Rate Limiting Restrictions.
13. In the **Skip Certificate Validation** box, specify if you want the IP to perform server certificate validation or not. This applies only when connecting to the server over HTTPS. When set to **True**, the IP will not perform certificate validation, typically used in secure environments, when working with trusted servers and self-signed certificates. When set to **False**, the IP will validate the server certificate. The server must be configured with a valid certificate signed by a valid certificate authority and the specified **Cisco Prime Infrastructure Server** name must be listed on the certificate.
14. Add additional connections if applicable.
15. Click **OK** to close the configuration dialog box, and then click **Finish**.

Additional Configuration

In addition to the configuration settings under the Options menu, you can further configure the integration pack using an XML formatted configuration file. The configuration file *Kelverion.IntegrationPack.Cisco.Pi.Configuration_3.10.xml* by default is installed at the same location as the IP assemblies.

On System Center Orchestrator 2019 or earlier:

[Program Files (x86)]\Common Files\Microsoft System Center 2012\Orchestrator\Extensions\Support\Integration Toolkit\022A33CF-0B57-4DF5-BBC2-FA21B625C8C4

On System Center Orchestrator 2022 or later:

[Program Files]\Common Files\Microsoft System Center 2012\Orchestrator\Extensions\Support\Integration Toolkit\022A33CF-0B57-4DF5-BBC2-FA21B625C8C4

Make sure to back up the configuration file before making changes.

The alarm and event Category/Condition values used internally by the Cisco Prime Infrastructure API are not always intuitive and they are inconsistent with the values displayed in the Cisco Prime Infrastructure web UI. The integration pack provides Category and Condition filter browsers which contain “display” values that are consistent with the Prime Infrastructure web UI. In the configuration XML file, you can specify how internal values map to display values (and vice versa).

If your Cisco Prime Infrastructure environment supports additional categories or conditions which are not part of the default installation, you can edit the configuration XML file to specify new mappings.

The category and conditions listed in Kelverion.IntegrationPack.Cisco.Pi.Configuration_3.10.xml are based on Cisco Prime Infrastructure version 3.10.

Adding a new category or condition in the configuration file will:

- Define Internal <=> Display mapping.
- Make that category/condition available in the filter browser list.

Adding a new entry in this configuration file will NOT define a new category/condition in Cisco Prime Infrastructure.

To add a new category, add a new <Category> element under **<AlarmCategories>** or **<EventCategories>**:

```
<Category display="New Category Display Name"
          internal="New Category Internal Name" />
```

To add a new condition, add a new <Condition> element under <AlarmConditions> or <EventCategories>:

```
<Condition display="New Condition Display Name"
           internal="New Condition Internal Name" />
```

The **display** attribute specifies the value that the IP will display in published data and filters. The **internal** attribute specifies the value used internally by Cisco Prime Infrastructure. To modify existing categories or conditions, simply find the category/condition in the list and modify the display or internal attributes accordingly.

Rate Limiting Restrictions

The Cisco Prime Infrastructure API implements rate limiting to protect the Prime Infrastructure server from overload situations caused by excessive requests. Please refer to Cisco documentation for more information and details on how you can configure rate limiting in your environment.

The IP takes API rate limiting into consideration and provides configuration settings which allow for adjusting the runtime behavior so that the IP tries to remain within the rate limiting restrictions and tries to recover when rate limiting errors are encountered.

- **Max Batch Results** controls the number of records returned in a single batch. This setting is particularly important when retrieving large numbers of records because it affects the total number of requests the IP will send in order to retrieve the entire record set. A low value means there will be more requests required to retrieve the entire record set. A large number means that the payload on each response (batch) will be larger since it contains a larger number of records.
- **Batch Interval (milliseconds)** controls how frequently the IP will send requests to the server when multiple batches are necessary to retrieve the entire record set. Ideally this value should be kept low, in order to minimize the amount of time waiting between requests, however, if this value is too low, then the IP may end requesting data too frequently and violate rate limiting restrictions.
- **Max Retry Count** specifies how many times the IP should retry a failed request in the case that a rate limiting error (*503 Service Unavailable*) is returned by the server. The IP implements a retry mechanism to handle rate limiting errors, however, its purpose is to

handle the odd occurrence when such an error is encountered, and the runbook should not be designed to rely on this mechanism for normal operations.

- **Retry Interval (seconds)** specifies how long the IP should wait between retries, when encountering rate limiting errors. Typically this value should not be too low, so that the system has a chance to recover in case rate limiting restrictions are violated.

In addition to the settings listed above, runbook design also plays an important role in making sure the IP stays within rate limit restrictions. *When configuring Prime Infrastructure runbooks, always be aware of situations where activities are executed in parallel, such as parallel branches or runbooks triggered multiple times by other runbooks.* In such cases, if parallel Prime Infrastructure activities run too often, it can potentially lead to a situation where rate limits are exceeded.

Cisco Prime Infrastructure Activities

This integration pack adds the KA Cisco Prime Infrastructure category to the **Activities** pane in the Client. This category contains the following activities:

- Get Access Points
- Get Alarms
- Get Devices
- Get Events
- Monitor Alarms
- Monitor Events

Common Configuration Instructions for All Activities

The following configuration instructions apply to all activities in this integration pack. Links to this section are included in the configuration instructions for each activity.

Activity Properties

Each activity has a set of required or optional properties that define the configuration of that activity. This includes how it connects to other activities or how the activity performs its actions. You can view or modify activity properties in the Orchestrator Client.

To configure the properties for an activity:

1. Double-click the activity. Alternatively, you can right-click the activity, and then click **Properties**.
2. To save your configuration entries, click **Finish**.

In the activity properties dialog box, several tabs along the left side provide access to general and specific settings for the activity. Although the number of available tabs for activity properties differs from activity to activity, all activities will have a **General** tab, a **Properties** tab and/or **Filters** tab, and a **Run Behavior** tab. Some activities may have additional tabs.

General Tab

This tab contains the **Name** and **Description** properties for the activity. By default, the **Name** of the activity is the same as its activity type, and the **Description** is blank. You can modify these properties to create more descriptive names or provide detailed descriptions of the actions of the activity.

Properties/Filters Tab

These tabs contain properties that are specific to the activity.

All activities in this integration pack have the **Configuration Name** property at the top of the **Properties** tab. This property is used to specify the connection to a Cisco Prime Infrastructure server.

To configure the Configuration Name property:

- Click the ellipsis **(…)** button next to the **Name** field, and then select the applicable connection name. Connections displayed in the list have been previously configured as described in Configuring the Integration Pack.

Filter Behavior

The Monitor and Get activities use filters to determine the values that will invoke a runbook or retrieve activities. Property values of potential candidates are compared to the values of the filters to determine if they meet the criteria. When matching against values, you select one of the available methods of comparison. An option is provided to either match or not match the filter using each method. For example, the "Does not" version of a method causes alerts that do not match the filter to trigger the runbook. The Monitor activity will only trigger if all filters match and the Get activity will only return Alarms that match all filters.

- **Equals**: the field of the record exactly matches the text or number specified in the filter.
- **Does not equal**: the field of the record does not exactly match the text or number specified in the filter.
- **Is less than**: the field of the record is less than the number specified in the filter.
- **Is less than or equal to**: the field of the record is less than or equal to the number specified in the filter.
- **Is greater than**: the field of the record is greater than the number specified in the filter.
- **Is greater than or equal to**: the field of the record is greater than or equal to the number specified in the filter.
- **Contains**: the field of the record contains the text specified in the filter.
- **Does not contain**: the field of the record does not contain the text specified in the filter.
- **Starts with**: the field of the record starts with the text specified in the filter.
- **Ends with**: the field of the record ends with the text specified in the filter.

Null Filter Behavior

When filtering by fields which can contain null values, records where those fields are null will not be included in the filtered result set. This primarily affects the behavior of non-inclusive filters such as "Does not equal" or "Does not contain".

To specifically return a set of records with specific null fields, you can specify $null as the filter value in the filter tab. *Note that this can only be used with **Equals** and **Does not equal** filters.*

For example, consider a system which contains 100 devices, 50 of those devices have Device Name starting with "Cisco" and 30 devices have Device Name null. The Get Devices activity will:

- Return 100 devices when no filters are specified.
- Return 50 devices when specifying the filter *Device Name contains Cisco*.
- Return 20 devices when specifying the filter *Device Name Does not contain Cisco*.
- Return 30 devices when specifying the filter *Device name Equals $null*.

Run Behavior Tab

This tab contains the properties that determine how the activity handles multi-value published data and what notifications will be sent if the activity fails or runs for an excessive period of time.

Multi-Value Published Data Behavior

The Get activities retrieve information from another activity or outside source and can return one or more values in the published data. For example, when you use the Get Collection Member activity, the data output from that activity might be a list of computers that belong to the specified collection.

By default, the data from the Get activity will be passed on as multiple individual outputs. This invokes the next activity as many times as there are items in the output. Alternatively, you can provide a single output for the activity by enabling the **Flatten** option. When you enable this option, you also choose a formatting option:

- **Separate with line breaks**. Each item is on a new line. This format is useful for creating human-readable text files for the output.
- **Separate with _** . Each item is separated by one or more characters of your choice.
- **Use CSV format**. All items are in CSV (comma-separated value) format. This format is useful for importing data into spreadsheets or other applications.

The activity will produce a new set of data every time it runs. The **Flatten** feature does not flatten data across multiple instances of the same activity.

Event Notifications

Some activities are expected to take a limited amount of time to complete. If they do not complete within that time they may be stalled or there may be another issue preventing them from completing. You can define the number of seconds to wait for completion of the action. After this period a platform event will be sent, and the issue will be reported. You can also choose whether to generate a platform event if the activity returns a failure.

To be notified when the activity takes longer than a specified time to run or fails to run:

1. In the **Event Notifications** box, enter the **number of seconds** of run time before a notification is generated.
2. Select **Report if activity fails to run** to generate run failure notifications.

For more information about Orchestrator events, see the “Event Notifications " topics in the Runbook Properties in the online documentation for System Center Orchestrator.

Published Data

Published data is the foundation of a working runbook. It is the data produced as a result of the actions of an activity. This data is published to an internal data bus that is unique for each runbook. Subsequent activities in the runbook can subscribe to this data and use it in their configuration. Link conditions also use this information to add decision-making capabilities to runbooks.

An activity can only subscribe to data from the activities that are linked before it in the runbook. You can use published data to automatically populate the property values needed by activities.

To use published data:

1. Right-click the property value box, click **Subscribe**, and then click **Published Data**.
2. Click the **Activity** drop-down box and select the activity from which you want to obtain the data.
3. To view additional data elements common to all activities, select **Show Common Published Data**.
4. Click the published data element that you want to use, and then click **OK**.

For a list of the data elements published by each activity, see the Published Data tables in the activity topic. For information about the common published data items, see the Published Data in the online documentation for System Center Orchestrator.

Get Access Points Activity

The **Get Access Points** activity is used in a runbook to retrieve and filter Prime Infrastructure access points.

When filtering by fields which can contain null values, records where those fields are null will not be included in the filtered result set. To filter by null values, you can specify **$null** as the filter value in **Equals** or **Does not equal** filters. For more details, please see Null Filter Behavior.

Required Properties

This activity does not provide any required properties.

Optional Properties

You can use the following properties, as necessary, to control how the activity runs.

Max Results	Specifies the maximum number of records to be returned by the activity.
Sort By	Specifies a property to be used for sorting the returned record set.
Sort Order	Specifies sort order.

Filters

This activity provides the following filters, that you can combine, to selectively filter which alerts to retrieve.

Access Point Name	Filter by *Access Point Name*.
Access Point Type	Filter by *Access Point Type*.
ID	Filter by access point *ID*.
IP Address	Filter by *IP Address*.
MAC Address	Filter by *MAC Address*.
Map Location	Filter by *Map Location*.
Model	Filter by *Model*.
Reachability Status	Filter by *Reachability Status*.
Serial Number	Filter by *Serial Number*.
Software Version	Filter by *Software Version*.

Published Data

This activity publishes the following activity-specified data.

Access Point Name	Access point name.
Access Point Type	Access point type.
Count	Number of records returned by the activity.

ID	Unique identifier for the access point.
IP Address	Access point IP address.
MAC Address	Base radio MAC address.
Map Location	SNMP location.
Model	Access point model.
Reachability Status	Indicates management availability or reachability of the managed network element. It can indicate the availability or reachability of the management agent serving as a proxy for the network element. Allowed values are: • UNKNOWN • REACHABLE • UNREACHABLE • AGENT_UNREACHABLE • AGENT_UNLOADED • PING_REACHABLE • PING_UNREACHABLE
Serial Number	Access point serial number.
Software Version	Access point software version.

Get Alarms Activity

The **Get Alarms** activity is used in a runbook to retrieve and filter Prime Infrastructure alarms.

When filtering by fields which can contain null values, records where those fields are null will not be included in the filtered result set. To filter by null values, you can specify **$null** as the filter value in **Equals** or **Does not equal** filters. For more details, please see Null Filter Behavior.

Required Properties

This activity does not provide any required properties.

Optional Properties

You can use the following properties, as necessary, to control how the activity runs.

Device Group	Specifies a device group for which alarms should be retrieved.
Max Results	Specifies the maximum number of records to be returned by the activity.
Sort By	Specifies a property to be used for sorting the returned record set.
Sort Order	Specifies sort order.
Virtual Domain	Specifies a virtual domain for which alarms should be retrieved.

Filters

This activity provides the following filters, that you can combine, to selectively filter which alerts to retrieve.

Acknowledged	Filter by *Acknowledged* value.
Alarm Found At	Filter by *Alarm Found At* value
Alarm Found At (UTC)	Filter by *Alarm Found At (UTC)* value
Alarm ID	Filter by *Alarm ID.*
Category	Filter by *Category.* Allowed values for this filter can be one of the following: a) Display Alarm Category values listed in the filter browser (internal values are also listed in parenthesis), which are mapped to internal values as specified in the configuration XML file. For details, see Additional Configuration. b) Internal Alarm Category values, which may or may not be specified in the configuration XML file, depending on your Prime Infrastructure environment. **Note**: When specifying a filter value which is not part of the filter browser list, the IP will bypass the display ←→ internal mapping and will treat the specified value as an **internal** value. For example, the IP maps the internal category value *AP,* to display value *Access Points*.

	The filters below contain values which will be matched internally to *AP*: *Category Equals Access Points (AP)* *Category Contains Access Points (AP)* *Category Does not equal AP* *Category Starts with A* *Category Ends with P* The filters below contain values which will **not** be matched internally to *AP*: *Category Contains Acc* *Category Starts with Access* *Category Ends with Points*
Condition	Filter by *Condition.* Allowed values for this filter can be one of the following: a) Display Alarm Condition values listed in the filter browser (internal values are also listed in parenthesis), which are mapped to internal values as specified in the configuration XML file. For details, see Additional Configuration. b) Internal Alarm Condition values, which may or may not be specified in the configuration XML file, depending on your Prime Infrastructure environment. **Note**: When specifying a filter value which is not part of the filter browser list, the IP will bypass the display ←→ internal mapping and will treat the specified value as an **internal** value. For example, the IP maps the internal condition value *SWT_SWITCH_DOWN,* to display value *Switch down*. The filters below contain values which will be matched internally to *SWT_SWITCH_DOWN*: *Condition Equals SWT_SWITCH_DOWN* *Condition Contains _SWITCH_* *Condition Does not equal Switch down* *Condition Starts with SWT* *Condition Ends with DOWN* The filters below contain values which will **not** be matched internally to *SWT_SWITCH_DOWN*: *Condition Contains Switch Do* *Condition Starts with Switch* *Condition Ends with down*
Device Name	Filter by *Device Name.*
Last Updated At	Filter by *Last Updated At* value.

Last Updated At (UTC)	Filter by *Last Updated At (UTC)* value.
Message	Filter by *Message.*
NTTY Address	Filter by *NTTY Address.*
Owner	Filter by *Owner.*
Severity	Filter by *Severity.*
Source	Filter by *Source.*
Wireless Specific Alarm ID	Filter by *Wireless Specific Alarm ID.*

Published Data

This activity publishes the following activity-specified data.

Acknowledged	Indicates whether the alarm has been acknowledged or not.
Alarm Found At	Specifies the time when this alarm was created. This is the same as the time of the event that resulted in the creation of this alarm.
Alarm Found At (UTC)	Specifies the time when this alarm was created, in UTC. This is the same as the time of the event that resulted in the creation of this alarm.
Alarm ID	A calculated opaque value that is used by the event handling implementation logic to identify the alarm and that serves as an identifier (sometimes known as a "business key"). In most cases, the specific alarm ID is an identifier for the entity that caused the alarm (known as the "alarm source" and available in the "Source" property) augmented by a distinguishing value (or values) to facilitate unique identifying value for this specific alarm. The distinguishing values can be, but are not limited to, the specific types of the alarm.
Category	Specifies a major category for the Alarm.
Condition	Specifies a type of occurrence/event within a category.
Count	Number of records returned by the activity.
Device Name	Represents the entity for which the event is reported.
Last Updated At	Date and time when the alarm record was last modified.
Last Updated At (UTC)	UTC date and time when the alarm record was last modified.
Message	Alarm message information.
NTTY Address	Used to associate AP alarms with their controller.
Owner	Specifies the name or ID of the person assigned to handle this alarm.
Severity	Specifies the alarm severity. The options are: • CRITICAL • MAJOR • MINOR

	• WARNING • CLEARED • INFORMATION
Source	Represents the entity for which the event/alarm is reported. This is not necessarily the reporting entity; in many cases, the notification is sent by an entity reporting an occurrence on another entity. Note that this is an opaque reference type that MIGHT be a reference to a known entity (that is, an instance in the model), but it might also be a reference to an entity that is not represented in the system.
Wireless Specific Alarm ID	ID used for associating Unified AP alarms with their Unified AP.

Get Devices Activity

The **Get Devices** activity is used in a runbook to retrieve and filter Prime Infrastructure devices.

When filtering by fields which can contain null values, records where those fields are null will not be included in the filtered result set. To filter by null values, you can specify **$null** as the filter value in **Equals** or **Does not equal** filters. For more details, please see Null Filter Behavior.

Required Properties

This activity does not provide any required properties.

Optional Properties

You can use the following properties, as necessary, to control how the activity runs.

Device Group	Specifies a device group for which devices should be retrieved.
Max Results	Specifies the maximum number of records to be returned by the activity.
Sort By	Specifies a property to be used for sorting the returned record set.
Sort Order	Specifies sort order.
Virtual Domain	Specifies a virtual domain for which devices should be retrieved.

Filters

This activity provides the following filters, that you can combine, to selectively filter which devices to retrieve.

Admin Status	Filter by *Admin Status.*
Collection Status	Filter by *Collection Status.*
Collection Detail	Filter by *Collection Detail.*
Collection Time	Filter by *Collection Time.*
Collection Time (UTC)	Filter by *Collection Time (UTC).*
Creation Time	Filter by *Creation Time.*
Creation Time (UTC)	Filter by *Creation Time (UTC).*
Device ID	Filter by *Device ID.*
Device Name	Filter by *Device Name.*
Device Type	Filter by *Device Type.*
ID	Filter by *ID.*
IP Address	Filter by *IP Address.*
Location	Filter by *Location.*
Management Status	Filter by *Management Status.*

Product Family	Filter by *Product Family.*
Reachability	Filter by *Reachability.*
Software Type	Filter by *Software Type.*
Software Version	Filter by *Software Version.*

Published Data

This activity publishes the following activity-specified data.

Admin Status	Represents the current admin status of the device. Allowed values: • UNMANAGED • MANAGED • MAINTENANCE
Collection Status	Last inventory collection status. Allowed values: • COMPLETED • MAJORCOMPLETED • COLLECTIONFAILURE • PARTIALCOLLECTIONFAILURE • SNMPCONNECTIVITYFAILED • WRONGCLICREDENTIALS • WRONGHTTPCREDENTIALS • SYNCHRONIZING • MAJORSYNCHRONIZING • MINORSYNCHRONIZING • SNMPUSERAUTHENTICATIONFAILED • NOLICENSE • ADDINITIATED • DELETEINPROGRESS • PINGUNREACHABLE • SPT_ONLY • IN_SERVICE_MAINTENANCE • IN_SERVICE
Collection Detail	Detailed status of inventory collection.
Collection Time	Date and time of the inventory collection.
Collection Time (UTC)	UTC date and time of the inventory collection.
Count	The number of records returned by this activity
Creation Time	Date and time when the instance of the device was created.
Creation Time (UTC)	UTC date and time when the instance of the device was created.
Device ID	An internal id to recognize the device, which is the id of the associated

	management network element associated with this device.
Device Name	The name of the device.
Device Type	The type of the device.
ID	Unique identifier for the device.
IP Address	The IP address of the device.
Location	The system location of the device
Management Status	Represents the current management state of the network element: managed, unmanaged, under maintenance, and so on. This state is modified by events in the network and network management system, and by user request. Options include: • UNKNOWN • ADDED_ININITIALSTATE • MANAGED_BUT_NEVERSYNCHRONIZED • MANAGED_AND_SYNCHRONIZED • MANAGED_BUT_OUTOFSYNC • MANAGED_BUT_LOSSOFCONNECTIVITY • PREPROVISIONED • UNMANAGED • INSERVICE_MAINTENANCE • MANAGED_BUT_INCOMPLETE • MANAGED_BUT_AGENTSHUTTINGDOWN • MANAGED_PREPARINGFORMAINTENANCE • MANAGED_BUT_DUPLICATE • MANAGED_BUT_CONFLICTINGCREDENTIALS • MANAGED_BUT_SYNCHRONIZING • UNMANAGED_UNLICENSED
Manufacturer Part Numbers	The device chassis details.
Product Family	The product family of this device.
Reachability	Indicates management availability or reachability of the managed network element. It can indicate the availability or reachability of the management agent serving as a proxy for the network element. Options include: • UNKNOWN • REACHABLE • UNREACHABLE • AGENT_UNREACHABLE • AGENT_UNLOADED • PING_REACHABLE

	- PING_UNREACHABLE
Software Type	A string that identifies the specific type of software that is installed. For example, Cisco IOS or Linux.
Software Version	The specific version of the software that is installed.

Get Events Activity

The **Get Events** activity is used in a runbook to retrieve and filter Prime Infrastructure events.

When filtering by fields which can contain null values, records where those fields are null will not be included in the filtered result set. To filter by null values, you can specify **$null** as the filter value in **Equals** or **Does not equal** filters. For more details, please see Null Filter Behavior.

Required Properties

This activity does not provide any required properties.

Optional Properties

You can use the following properties, as necessary, to control how the activity runs.

Device Group	Specifies a device group for which events should be retrieved.
Max Results	Specifies the maximum number of records to be returned by the activity.
Sort By	Specifies a property to be used for sorting the returned record set.
Sort Order	Specifies sort order.
Virtual Domain	Specifies a virtual domain for which events should be retrieved.

Filters

This activity provides the following filters, that you can combine, to selectively filter which events to retrieve.

Category	Filter by *Category.* Allowed values for this filter can be one of the following: a) Display Event Category values listed in the filter browser (internal values are also listed in parenthesis), which are mapped to internal values as specified in the configuration XML file. For details, see Additional Configuration. b) Internal Event Category values, which may or may not be specified in the configuration XML file, depending on your Prime Infrastructure environment. **Note**: When specifying a filter value which is not part of the filter browser list, the IP will bypass the display ←→ internal mapping and will treat the specified value as an **internal** value. For example, the IP maps the internal category value *AP,* to display value *Access Points*. The filters below contain values which will be matched internally to *AP*: *Category Equals Access Points (AP)* *Category Contains Access Points (AP)* *Category Does not equal AP* *Category Starts with A* *Category Ends with P*

	The filters below contain values which will **not** be matched internally to *AP*: *Category Contains Acc* *Category Starts with Access* *Category Ends with Points*
Condition	Filter by *Condition.* Allowed values for this filter can be one of the following: a) Display Event Condition values listed in the filter browser (internal values are also listed in parenthesis), which are mapped to internal values as specified in the configuration XML file. For details, see Additional Configuration. b) Internal Event Condition values, which may or may not be specified in the configuration XML file, depending on your Prime Infrastructure environment. **Note**: When specifying a filter value which is not part of the filter browser list, the IP will bypass the display ←→ internal mapping and will treat the specified value as an **internal** value. For example, the IP maps the internal condition value *SWT_SWITCH_DOWN,* to display value *Switch down*. The filters below contain values which will be matched internally to *SWT_SWITCH_DOWN*: *Condition Equals SWT_SWITCH_DOWN* *Condition Contains _SWITCH_* *Condition Does not equal Switch down* *Condition Starts with SWT* *Condition Ends with DOWN* The filters below contain values which will **not** be matched internally to *SWT_SWITCH_DOWN*: *Condition Contains Switch Do* *Condition Starts with Switch* *Condition Ends with down*
Correlated	Filter by *Correlated* value.
Description	Filter by *Description.*
Device Name	Filter by *Device Name.*
Event Found At	Filter by *Event Found At* value.
Event Found At (UTC)	Filter by *Event Found At (UTC)* value.
Event ID	Filter by *Event ID.*
Severity	Filter by *Severity.*
Source	Filter by *Source.*

Published Data

This activity publishes the following activity-specified data.

Category	Specifies a major category for the event.
Condition	Specifies a type of occurrence/event within a category.
Correlated	Correlated value.
Count	Number of records returned by the activity.
Description	Event description.
Device Name	Represents the entity for which the event is reported.
Event Found At	Specifies the time the event occurred. If this time is not available (because the raw notification did not carry this information), the timestamp specifies the time this record was created.
Event Found At (UTC)	Specifies the time the event occurred, in UTC. If this time is not available (because the raw notification did not carry this information), the timestamp specifies the time this record was created.
Event ID	Unique event identifier.
Severity	Specifies the event severity. Valid values are: • CRITICAL • MAJOR • MINOR • WARNING • CLEARED • INFORMATION
Source	Represents the entity for which the event/alarm is reported. This is not necessarily the reporting entity; in many cases, the notification is sent by an entity reporting an occurrence on another entity. Note that this is an opaque reference type that MIGHT be a reference to a known entity (that is, an instance in the model), but it might also be a reference to an entity that is not represented in the system.

Monitor Alarms Activity

The **Monitor Alarms** activity is used in a runbook to detect new and/or updated Prime Infrastructure alarms.

When filtering by fields which can contain null values, records where those fields are null will not be included in the filtered result set. To filter by null values, you can specify **$null** as the filter value in **Equals** or **Does not equal** filters. For more details, please see Null Filter Behavior.

Required Properties

You must configure the following properties.

Monitor Interval	Specifies the number of seconds the monitor waits between poll requests. Minimum value is 15 seconds.
Monitor New	Specifies whether the monitor will detect new alarms, which have not previously been raised in Prime Infrastructure.
Monitor Updated	Specifies whether the monitor will detect changes to existing alarms, which have previously been raised in Prime Infrastructure and are now being modified.

Optional Properties

You can use the following properties, as necessary, to control how the activity runs.

Device Group	Specifies a device group for which alarms should be retrieved.
Sort By	Specifies a property to be used for sorting the returned record set.
Sort Order	Specifies sort order.
Virtual Domain	Specifies a virtual domain for which alarms should be retrieved.

Filters

This activity provides the following filters, that you can combine, to selectively filter which alerts will trigger the monitor.

Acknowledged	Filter by *Acknowledged* value.
Alarm ID	Filter by *Alarm ID.*
Category	Filter by *Category.* Allowed values for this filter can be one of the following: a) Display Alarm Category values listed in the filter browser (internal values are also listed in parenthesis), which are mapped to internal values as specified in the configuration XML file. For details, see Additional Configuration. b) Internal Alarm Category values, which may or may not be specified in the configuration XML file, depending on your Prime Infrastructure environment.

	Note: When specifying a filter value which is not part of the filter browser list, the IP will bypass the display ←→ internal mapping and will treat the specified value as an **internal** value. For example, the IP maps the internal category value *AP,* to display value *Access Points*. The filters below contain values which will be matched internally to *AP*: *Category Equals Access Points* *Category Contains Access Points* *Category Does not equal AP* *Category Starts with A* *Category Ends with P* The filters below contain values which will **not** be matched internally to *AP*: *Category Contains Acc* *Category Starts with Access* *Category Ends with Points*
Condition	Filter by *Condition.* Allowed values for this filter can be one of the following: a) Display Alarm Condition values listed in the filter browser (internal values are also listed in parenthesis), which are mapped to internal values as specified in the configuration XML file. For details, see Additional Configuration. b) Internal Alarm Condition values, which may or may not be specified in the configuration XML file, depending on your Prime Infrastructure environment. **Note***: When specifying a filter value which is not part of the filter browser list, the IP will bypass the display ←→ internal mapping and will treat the specified value as an* ***internal*** *value.* For example, the IP maps the internal condition value *SWT_SWITCH_DOWN,* to display value *Switch down*. The filters below contain values which will be matched internally to *SWT_SWITCH_DOWN*: *Condition Equals SWT_SWITCH_DOWN* *Condition Contains _SWITCH_* *Condition Does not equal Switch down* *Condition Starts with SWT* *Condition Ends with DOWN* The filters below contain values which will **not** be matched internally to *SWT_SWITCH_DOWN*: *Condition Contains Switch Do* *Condition Starts with Switch* *Condition Ends with down*

Device Name	Filter by *Device Name.*
ID	Filter by *ID.*
Message	Filter by *Message.*
NTTY Address	Filter by *NTTY Address.*
Owner	Filter by *Owner.*
Severity	Filter by *Severity.*
Source	Filter by *Source.*
Time Stamp	Filter by *Time Stamp* value.
Wireless Specific Alarm ID	Filter by *Wireless Specific Alarm ID.*

Published Data

This activity publishes the following activity-specified data.

Acknowledged	Indicates whether the alarm has been acknowledged or not.
Alarm Found At	Specifies the time when this alarm was created. This is the same as the time of the event that resulted in the creation of this alarm.
Alarm Found At (UTC)	Specifies the time when this alarm was created, in UTC. This is the same as the time of the event that resulted in the creation of this alarm.
Alarm ID	A calculated opaque value that is used by the event handling implementation logic to identify the alarm and that serves as an identifier (sometimes known as a "business key"). In most cases, the specific Alarm ID is an identifier for the entity that caused the alarm (known as the "alarm source" and available in the "Source" property) augmented by a distinguishing value (or values) to facilitate unique identifying value for this specific alarm. The distinguishing values can be, but are not limited to, the specific types of the alarm.
Category	Specifies a major category for the Alarm.
Condition	Specifies a type of occurrence/event within a category.
Count	Number of records returned by the activity.
Device Name	Represents the entity for which the event is reported.
Last Updated At	Date and time when the alarm record was last modified.
Last Updated At (UTC)	UTC date and time when the alarm record was last modified.
Message	Alarm message information.
NTTY Address	Used to associate AP alarms with their controller.
Owner	Specifies the name or ID of the person assigned to handle this alarm.

Severity	Specifies the alarm severity. Valid values are: • CRITICAL • MAJOR • MINOR • WARNING • CLEARED • INFORMATION
Source	Represents the entity for which the event/alarm is reported. This is not necessarily the reporting entity; in many cases, the notification is sent by an entity reporting an occurrence on another entity. Note that this is an opaque reference type that MIGHT be a reference to a known entity (that is, an instance in the model), but it might also be a reference to an entity that is not represented in the system.
Wireless Specific Alarm ID	ID used for associating Unified AP alarms with their Unified AP.

Monitor Events Activity

The **Monitor Events** activity is used in a runbook to detect new Prime Infrastructure events.

When filtering by fields which can contain null values, records where those fields are null will not be included in the filtered result set. To filter by null values, you can specify **$null** as the filter value in **Equals** or **Does not equal** filters. For more details, please see Null Filter Behavior.

Required Properties

You must configure the following properties.

Monitor Interval	Specifies the number of seconds the monitor waits between poll requests. Minimum value is 15 seconds.

Optional Properties

You can use the following properties, as necessary, to control how the activity runs.

Device Group	Specifies a device group for which events should be retrieved.
Sort By	Specifies a property to be used for sorting the returned record set.
Sort Order	Specifies sort order.
Virtual Domain	Specifies a virtual domain for which events should be retrieved.

Filters

This activity provides the following filters, that you can combine, to selectively filter which events will trigger the monitor.

Category	Filter by *Category.* Allowed values for this filter can be one of the following: a) Display Event Category values listed in the filter browser (internal values are also listed in parenthesis), which are mapped to internal values as specified in the configuration XML file. For details, see Additional Configuration. b) Internal Event Category values, which may or may not be specified in the configuration XML file, depending on your Prime Infrastructure environment. **Note**: *When specifying a filter value which is not part of the filter browser list, the IP will bypass the display ←→ internal mapping and will treat the specified value as an* ***internal*** *value.* For example, the IP maps the internal category value *AP,* to display value *Access Points*. The filters below contain values which will be matched internally to *AP*: *Category Equals Access Points (AP)* *Category Contains Access Points (AP)* *Category Does not equal AP*

	Category Starts with A *Category Ends with P* The filters below contain values which will **not** be matched internally to *AP*: *Category Contains Acc* *Category Starts with Access* *Category Ends with Points*
Condition	Filter by *Condition.* Allowed values for this filter can be one of the following: a) Display Event Condition values listed in the filter browser (internal values are also listed in parenthesis), which are mapped to internal values as specified in the configuration XML file. For details, see Additional Configuration. b) Internal Event Condition values, which may or may not be specified in the configuration XML file, depending on your Prime Infrastructure environment. **Note**: *When specifying a filter value which is not part of the filter browser list, the IP will bypass the display ←→ internal mapping and will treat the specified value as an **internal** value.* For example, the IP maps the internal condition value *SWT_SWITCH_DOWN,* to display value *Switch down*. The filters below contain values which will be matched internally to *SWT_SWITCH_DOWN*: *Condition Equals SWT_SWITCH_DOWN* *Condition Contains _SWITCH_* *Condition Does not equal Switch down* *Condition Starts with SWT* *Condition Ends with DOWN* The filters below contain values which will **not** be matched internally to *SWT_SWITCH_DOWN*: *Condition Contains Switch Do* *Condition Starts with Switch* *Condition Ends with down*
Correlated	Filter by *Correlated* value.
Description	Filter by *Description.*
Device Name	Filter by *Device Name.*
Event ID	Filter by *Event ID.*
Severity	Filter by *Severity.*
Source	Filter by *Source.*

Published Data

This activity publishes the following activity-specified data.

Category	Specifies a major category for the event.
Condition	Specifies a type of occurrence/event within a category.
Correlated	Correlated value.
Count	Number of records returned by the activity.
Description	Event description.
Device Name	Represents the entity for which the event is reported.
Event Found At	Date and time when the event was found.
Event Found At (UTC)	UTC date and time when the event was found.
Event ID	Unique event identifier.
Severity	Specifies the event severity. Valid values are: • CRITICAL • MAJOR • MINOR • WARNING • CLEARED • INFORMATION
Source	Represents the entity for which the event/alarm is reported. This is not necessarily the reporting entity; in many cases, the notification is sent by an entity reporting an occurrence on another entity. Note that this is an opaque reference type that MIGHT be a reference to a known entity (that is, an instance in the model), but it might also be a reference to an entity that is not represented in the system.