

How it works

My new Honeycomb-ServiceNow Integration allows you to:

- **Create or update Configuration Items** in your ServiceNow CMDB for entities that have recent data in Honeycomb. That includes application/parent services and underlying applications today (and I'm planning to add more classes soon).
- **Create rich Service Maps** in ServiceNow for the systems you're observing in Honeycomb. These maps are based on real traffic between CI's and are periodically updated based on the latest telemetry data.

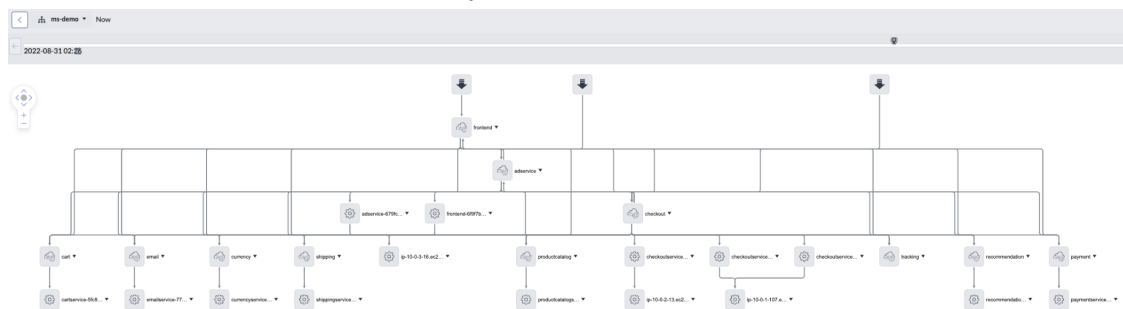


Figure 1: an example service map as generated by my new integration. If you're using our Kubernetes operator or OpenTelemetry's Kubernetes tags, you'll even see pod and node entities and relationships based on your trace data

- **Observe the impacts of Change Requests directly in Honeycomb.** For change requests logged in ServiceNow against entities being observed in Honeycomb, you can automatically launch straight into Honeycomb in the right context to see service performance for the change-affected resources: both before and after the change was implemented. The "Open in Honeycomb" button allows you to validate if the service has any negative impacts from the change in real-time!

Change Request
CHG0030171

Scheduling conflict detected. Use the [Scheduling Assistant](#) to avoid conflicts.

New ✓ Assess ✓ Authorize ✓ Scheduled ✓ Implement Review Closed Canceled

Number: CHG0030171
Requested by: Matt Morris
Category: Other
Service:
Service offering:
Configuration item: checkout
Priority: 4 - Low
Risk: Moderate
Impact: 3 - Low
Short description: Merge PR #46 to checkoutservice
Description: <https://github.com/honeycombio/microservices-demo/commit/5645075>

Model: Normal
Type: Normal
State: Implement
On hold: ☐
Conflict status: Conflict
Conflict last run: 2022-09-06 16:50:54
Assignment group: Cloud Operator Group
Assigned to:
CAB required: ☐
CAB date:
Actual start date: 2022-09-06 16:50:53
Actual end date:
CAB delegate:
CAB recommendation:
Review Conflict Calendar Update Open in Honeycomb Force to Update Set Delete

Planning Schedule Conflicts Notes Closure Information

Planned start date and Planned end date are the approved change window

Planned start date: 2022-09-06 16:49:10
Planned end date: 2022-09-06 18:49:16
CAB required: ☐
CAB date:
Actual start date: 2022-09-06 16:50:53
Actual end date:
CAB delegate:
CAB recommendation:
Review Conflict Calendar Update Open in Honeycomb Force to Update Set Delete

Figure 2: the “Open in Honeycomb” button allows you to open a dashboard in Honeycomb directly from a Change Request

- **Create rich Incidents in ServiceNow for Honeycomb Triggers and Honeycomb SLOs.** You can add your ServiceNow instance as a recipient on any Trigger or SLO burn alert, and incidents will be created when the specified conditions are met. ServiceNow incidents will be automatically updated if the trigger condition in Honeycomb is no longer met or if the SLO burn alert status changes to green. When used in this setting, the “Open in Honeycomb” button opens these in context for rapid troubleshooting!

Incident - INC0024028

Number: INC0024028
Caller: Honeycomb Integration
Category: Inquiry / Help
Subcategory: -- None --
Service: ms-demo
Service offering:
Configuration item: frontend
Short description: Resolved: 'User Latency' will not violate SLO in BHCmOs
Description:

```
{
  "version": "v0.1.0",
  "name": "User Latency",
  "id": "CYdgk9KDAu5",
  "status": "OK",
  "summary": "Resolved: 'User Latency' will not violate SLO in BHCmOs",
  "description": "The error budget for SLO 'User Latency' is no longer in danger of being exhausted",
  "is_test": false
}
```

Channel: -- None --
State: Closed
Impact: 2 - Medium
Urgency: 2 - Medium
Priority: 3 - Moderate
Assignment group:
Assigned to:
Update Open in Honeycomb Resolve Delete

Notes Related Records Resolution Information

Problem:
Change Request:
Caused by Change:
Update Open in Honeycomb Resolve Delete

Related Links
Create Improvement Initiative
Create Special Handling Notes
Show in Honeycomb
Show SLA Timeline

Figure 3: the “Open in Honeycomb” button also works on Incidents from Burn Alerts and can drop you directly into troubleshooting your SLO from ServiceNow

- **Open any Incident directly in Honeycomb that is affecting an entity you are observing in Honeycomb.** Go straight to the affected resources during the right time frame, and jumpstart your troubleshooting process.

Figure 4: even incidents that weren't created by Honeycomb Triggers or SLOs will allow you to open a relevant query in Honeycomb if the affected CI/service on the Incident is being observed by Honeycomb

Who this integration is for

My hope for my new integration is that anyone and everyone can find some use for this! But it does leverage features in Honeycomb and ServiceNow that require certain paid features. In order to use this you'll need the following requirements:

- A [Honeycomb Enterprise](#) subscription
- A ServiceNow Flow Designer subscription that allows REST calls to external endpoints

If you have those two things, you're all set! You don't need to deploy any additional agents, further scale your ServiceNow deployment, or any additional subscriptions. Most notably:

- No ServiceNow MID Server is required
- No ServiceNow ITOM subscription is required

How to get started

The process of setting up my new integration is as easy as it gets. First, install the integration and add one or more API keys for your environments. Then, attach your ServiceNow instance as a recipient for any [Honeycomb Triggers](#) and/or [Honeycomb SLOs](#) that you want to send Incidents to ServiceNow.

Detailed steps for doing that are provided below:

1. Download the [update set](#) to your local drive
2. Import, preview, and commit the update set in ServiceNow
 - a. In ServiceNow, navigate to “Retrieved Update Sets”, scroll to the bottom of the window, and click “Import Update Set from XML.” Select the update set you downloaded

Related Links

[Import Update Set from XML](#)

- b. Locate the update set in the window you are redirected to and open it
- c. Click “Preview Update Set” on the top right of the window



- d. If there are any warnings or errors, scroll down and locate the relevant tab. Select all warnings or errors, and choose “accept remote”
- e. Click “Commit Update Set” on the top right of the window

3. Create API keys (optional)
 - a. If they don’t yet exist, you’ll need to create one [API key](#) for each Honeycomb environment that you want to integrate with ServiceNow. For each API key, please ensure it has the following permissions in Honeycomb: “Manage Queries and Columns”, “Run Queries”, and “Manage Recipients”

Edit API Key

Name

sn-integration

☒ Visible to Team Members
When unchecked, your API Key will be visible to Team Owners only.

☒ Enable
When unchecked, your API key will deactivate and your data will be rejected.

What should this API key be allowed to do?
[Read more](#) about API Key permissions

☐ Send Events
Send events to Honeycomb.

☐ Create Datasets
Create, edit, or delete Datasets. Datasets may be created 1) explicitly with the Datasets API or 2) implicitly when sending events to a non-existent Dataset.

☒ Manage Queries and Columns
Create, edit, or delete queries, columns, Derived Columns, and query annotations.

☒ Run Queries
Execute existing queries.

☐ Manage Public Boards
Create, edit, or delete Boards that are set to 'public'.

☐ Manage SLOs
Create, edit, or delete SLOs and their Burn Alerts.

☐ Manage Triggers
Create, edit, or delete Triggers.

☒ Manage Recipients
Create, edit, or delete recipients for Triggers and Burn Alerts.

☐ Manage Markers
Create, edit, or delete Markers.

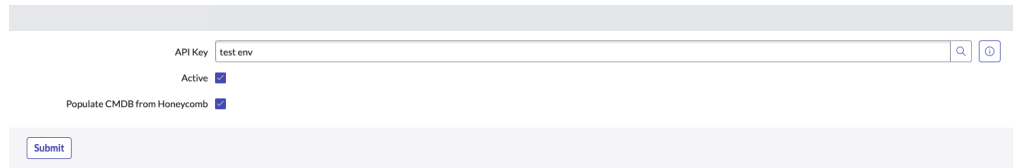
Changes may take up to a minute to propagate

Cancel

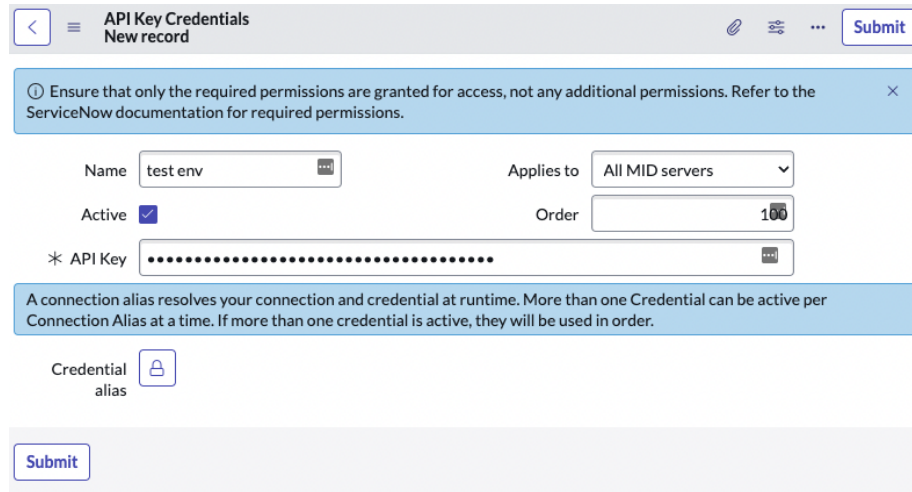
Save

4. Input the API key into ServiceNow for each environment you want to integrate
 - a. Navigate to Honeycomb Integration -> “Honeycomb Environments”

- b. Click “New” on the top right



- c. Click the magnifying glass next to the field “API Key”
d. If the relevant Honeycomb API Key has already been created as a record in ServiceNow, select it from the list. Otherwise, click “New” at the top right of the window listing API Keys



- e. If you don't want to import observed entities from Honeycomb into your ServiceNow CMDB, uncheck “**Populate CMDB from Honeycomb**”
f. Click “Submit” on the Honeycomb Environment record
5. Add your ServiceNow instance as a recipient for alerts on SLOs and/or Triggers in Honeycomb
- a. In Honeycomb, on any Trigger or SLO, you now have the option of adding your ServiceNow instance as a recipient

- i. To send the SLO Burn Alert or Trigger Alert to ServiceNow as an event (`em_event` table), use the recipient called “ServiceNow Event”

Add Trigger Recipient

Recipient

Webhook – ServiceNow Incident - aiopsdemo

Type: Webhook

Name: ServiceNow Incident -

URL: <https://now.com/api/snc/honeycomb/alert>

Shared Secret:

Note: Webhook recipients can only be edited in the [integration center](#).

Cancel

Add

- ii. To send the SLO Burn Alert or Trigger Alert to ServiceNow as an incident (`incident` table), use the recipient called “ServiceNow Incident”

Create Burn Alert

Exhaustion time (hours)

8

Notify

Webhook – ServiceNow Incident -

Cancel

Create Burn Alert

- b. If you want to send custom attributes with the webhook, format them as comma-separated, key-value pairs in the description of the Trigger or SLO definition. If you pass in `assignment_group`, `cmdb_ci`, or `severity` as below, they will be automatically handled as such in ServiceNow on the resulting SN Alert or Incident.

Edit Trigger

Name (Required)

Frontend latency

Give this trigger a name to describe what happened

Description (Optional)

assignment_group=Cloud Operator Group,severity=Critical,cmdb_ci=ms-demo

Your chance to provide next steps/followups; will be displayed in notifications when the trigger fires

Update SLO

[Learn more about creating SLOs.](#)

Name

Product View Latency

Description

assignment_group=Cloud Operator Group,severity=Critical

6. Review results
- a. Type “Honeycomb” into the filter navigator

b. Click “Dashboard”



Try it today

This new Honeycomb-ServiceNow integration is my own personal project and a community contribution. Although I work for Honeycomb, I'm also a part of the Pollinators community and I'm thrilled to be able to share this experiment with you. My hope is that you find it useful in your workflows and that, together, we can figure out how to make this better and even more useful.

Let me know what you think. Feel free to hit me with questions, comments, or any issues you run into. You can reach me on Pollinators or at mattmorris@honeycomb.io.

Cheers!