Beyond Identity Events Data integration with Elastic – PUSH

1 Table of Contents

2 Introduction

2.1 About

This guide provides instructions on how to:

- Integrate BI events data with Elastic. Elastic supports events push and events pull models

2.2 Prerequisites

Ensure that you have the following:

- You have a tenant configured for your organization and able to enroll users.
- You have an Elastic cloud account with admin privilege

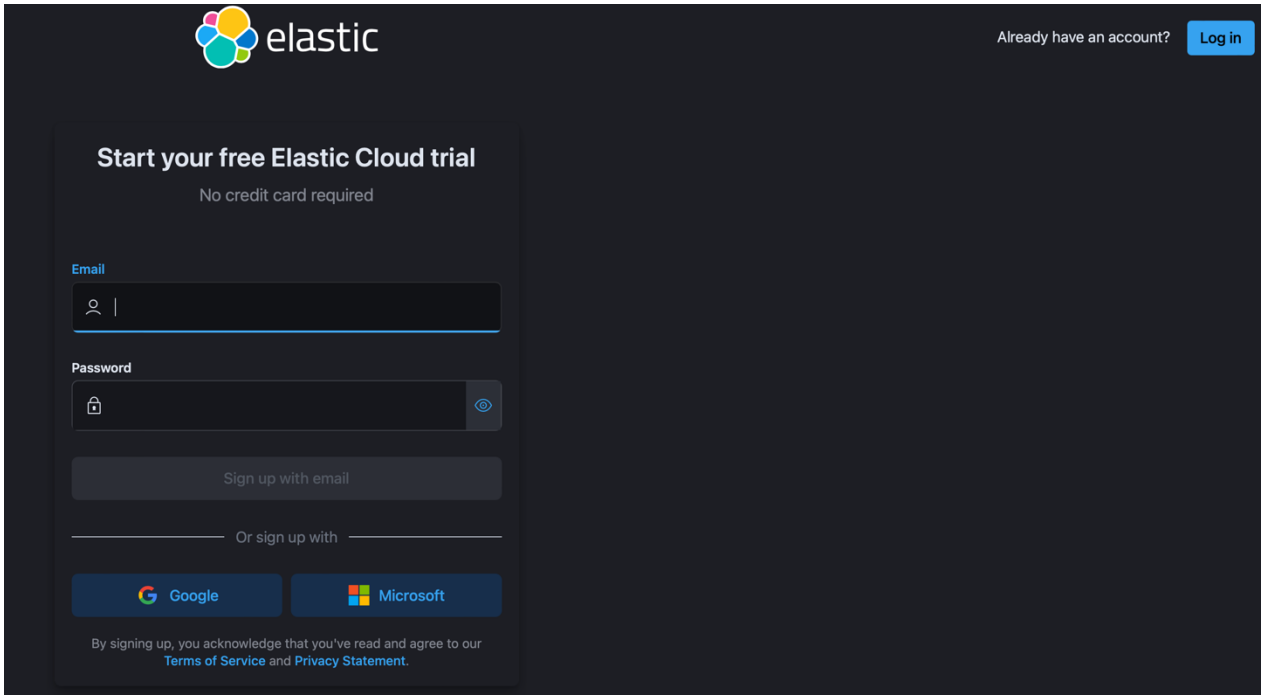
- The Firewall ports should be open to allow Beyond Identity[BI] to push events to your elastic deployment. Reach out to your BI contact for the ports details

3 Elastic configuration

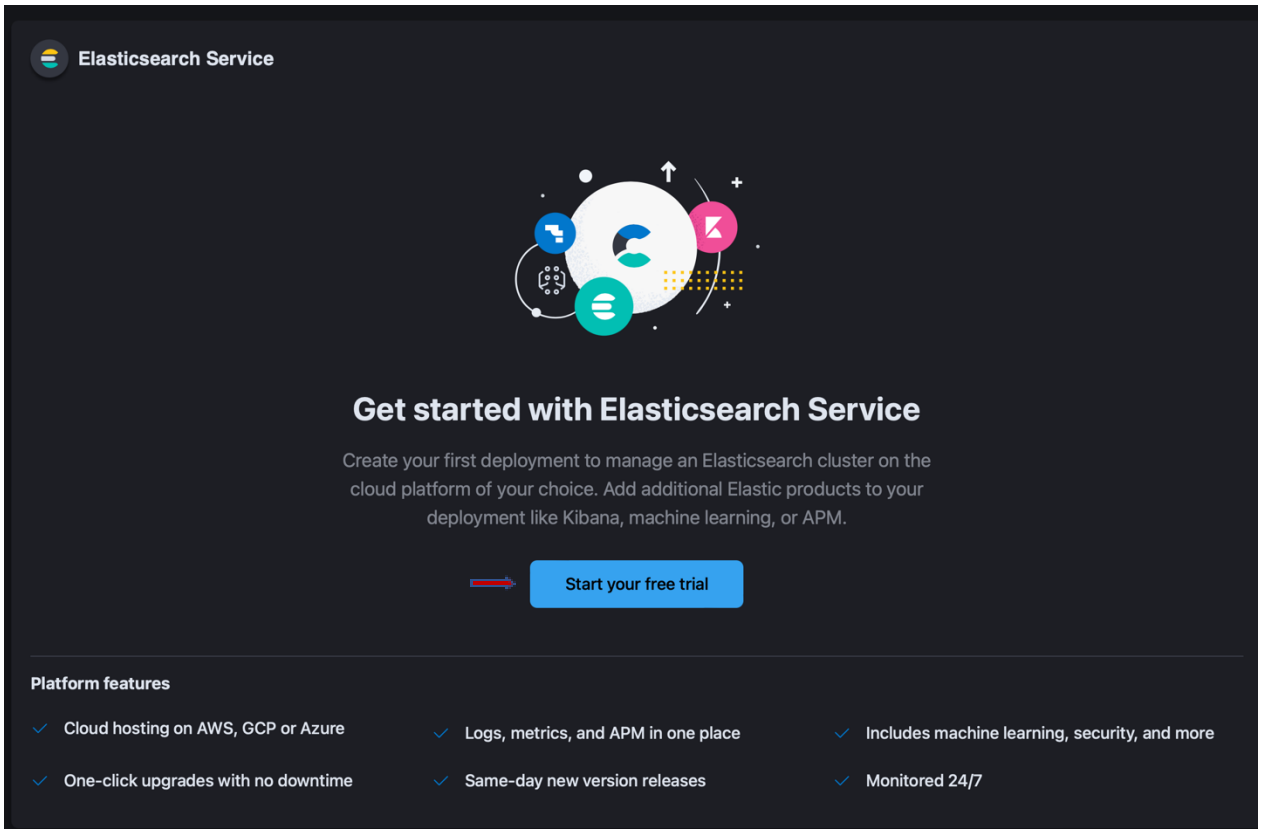
- Create Elastic deployment
- Install Agent, Add to fleet
- Create Agent Policy / Add agent
- Add HTTP Logs Integration to agent policy [Push]
- Push tenant events to Elastic

3.1 Create Elastic deployment

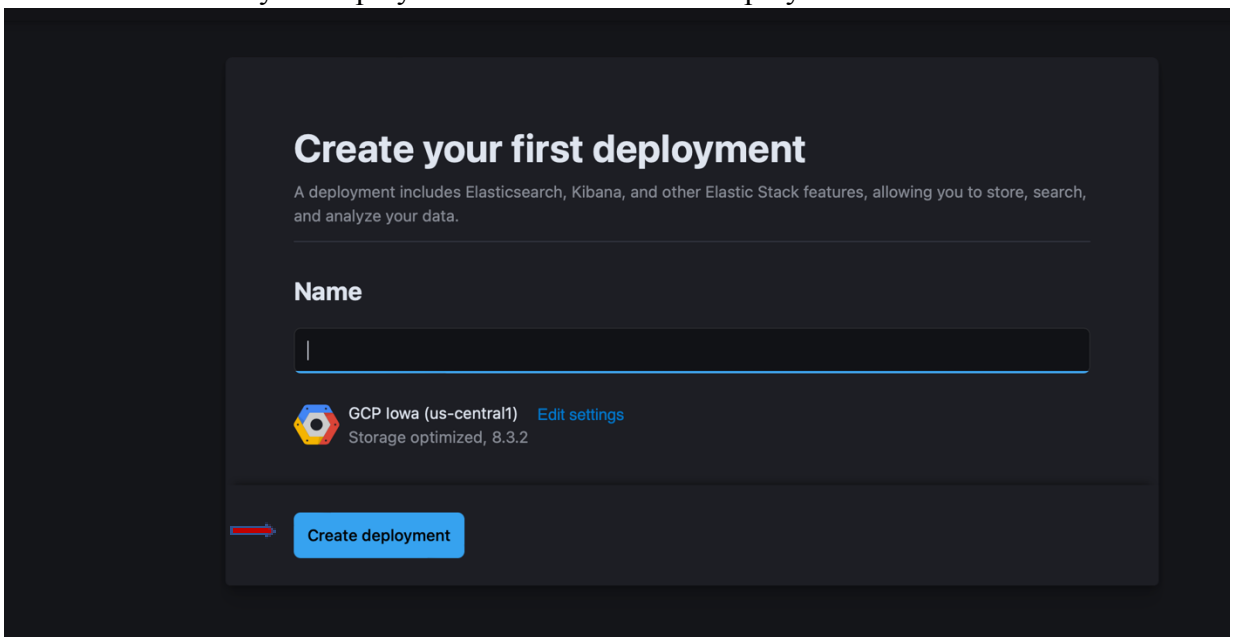
You need an Elastic account to create a deployment and configure push or pull. You can start free by accessing https://cloud.elastic.co/registration and sign up for a 14-day trial account



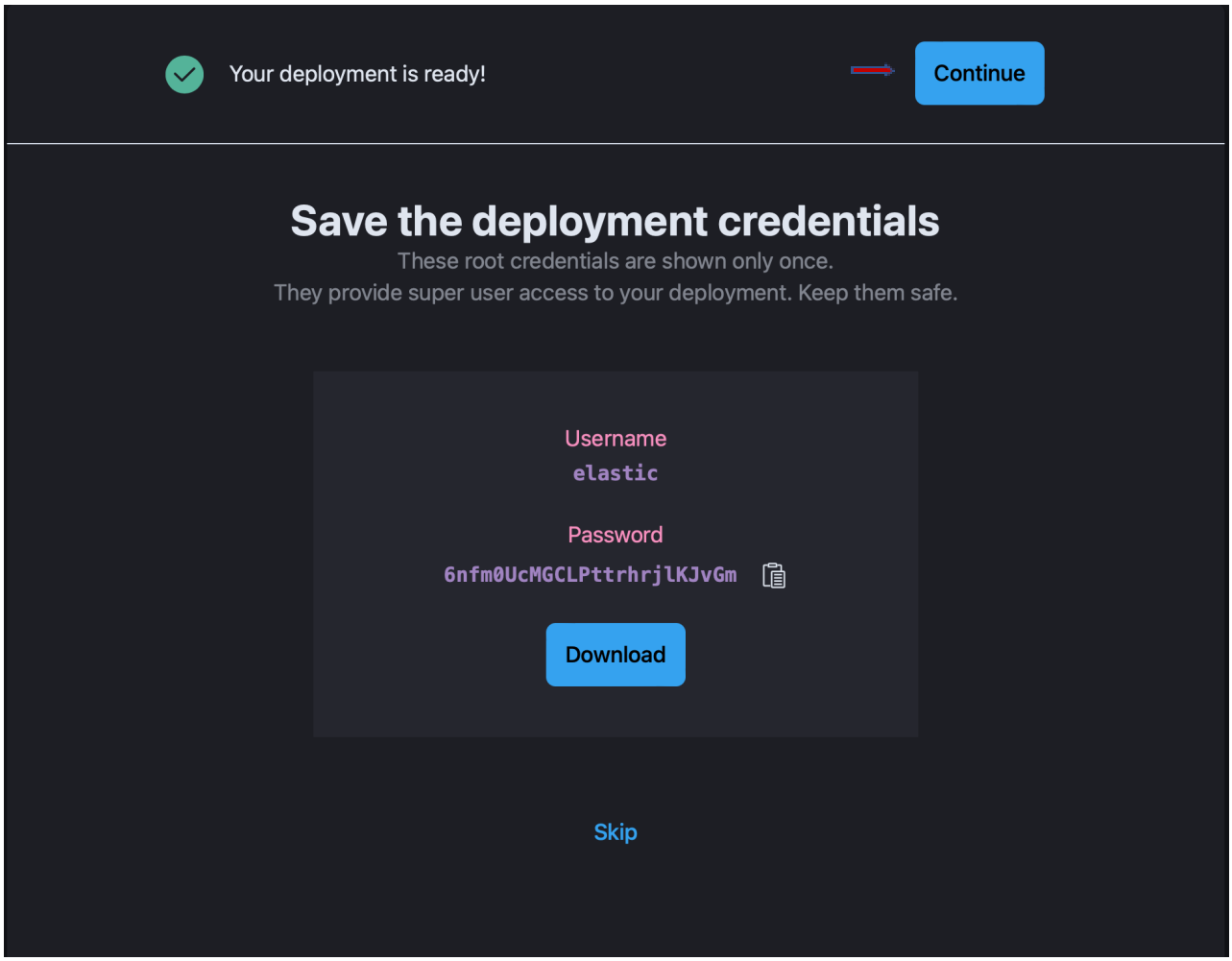
Once you the account is created, click on “Start your free trial”



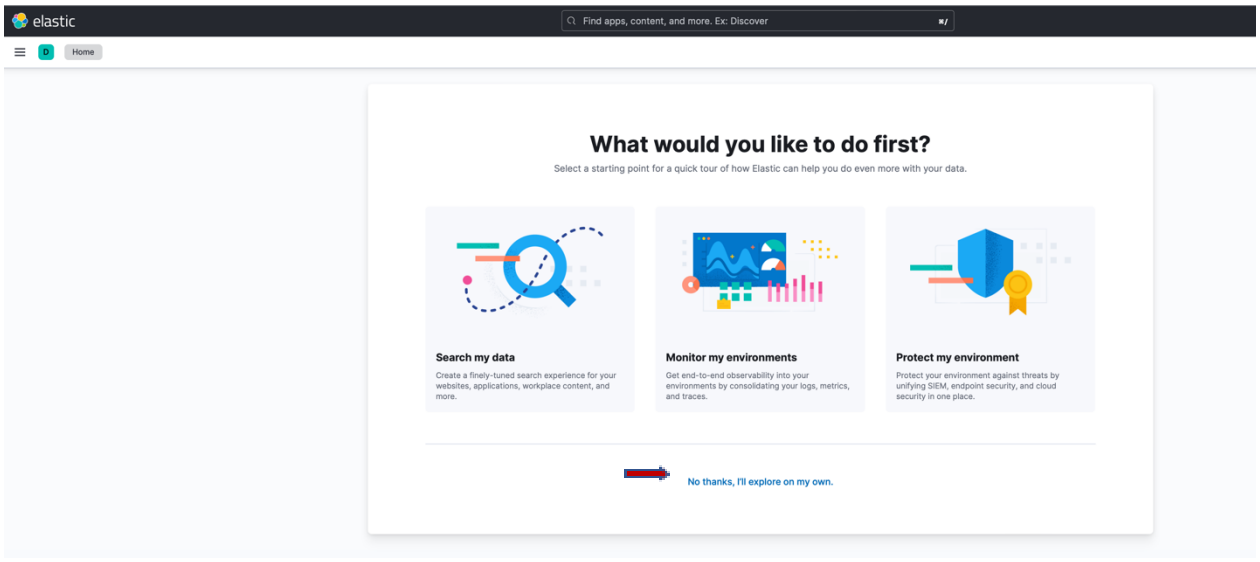
Provide a Name for your deployment and click “Create deployment”.

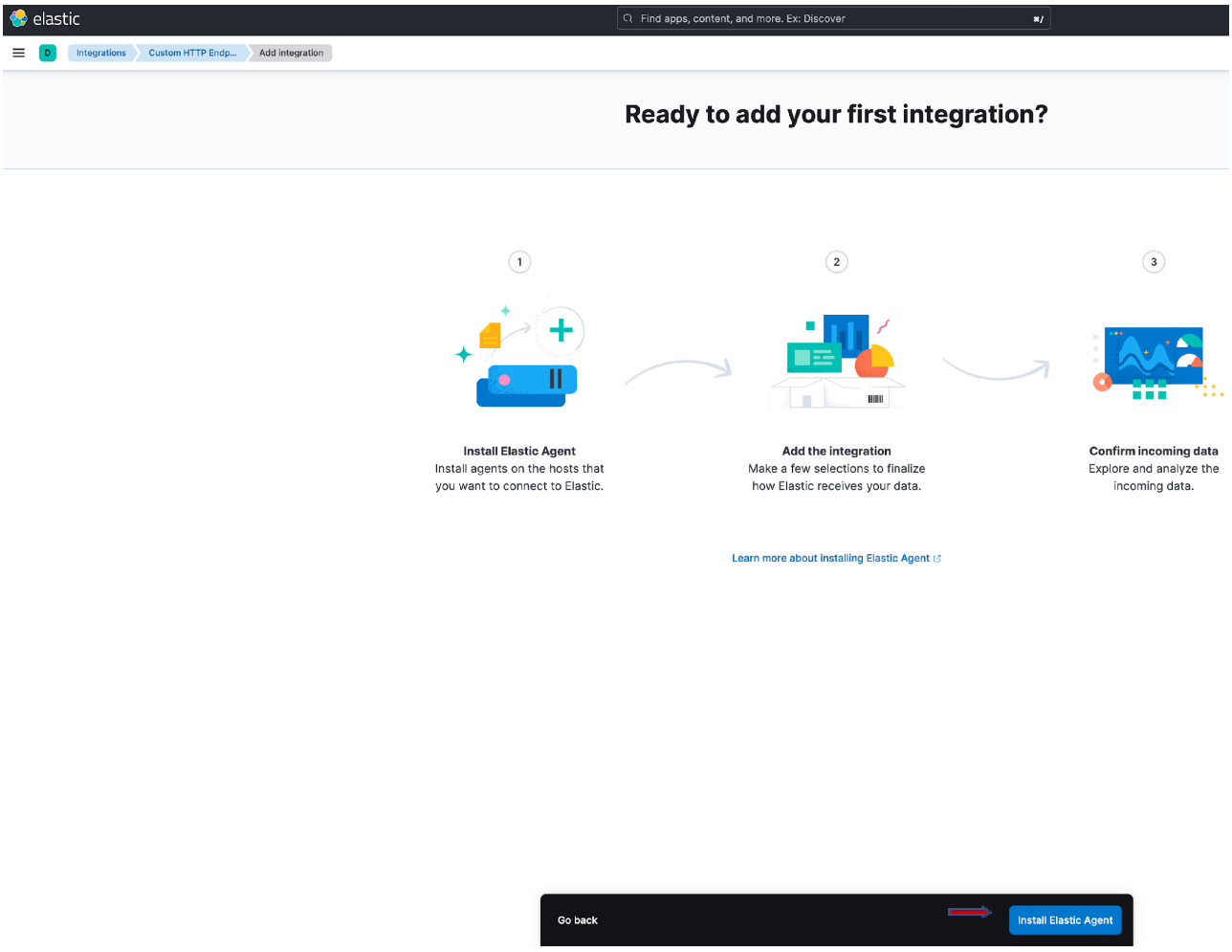


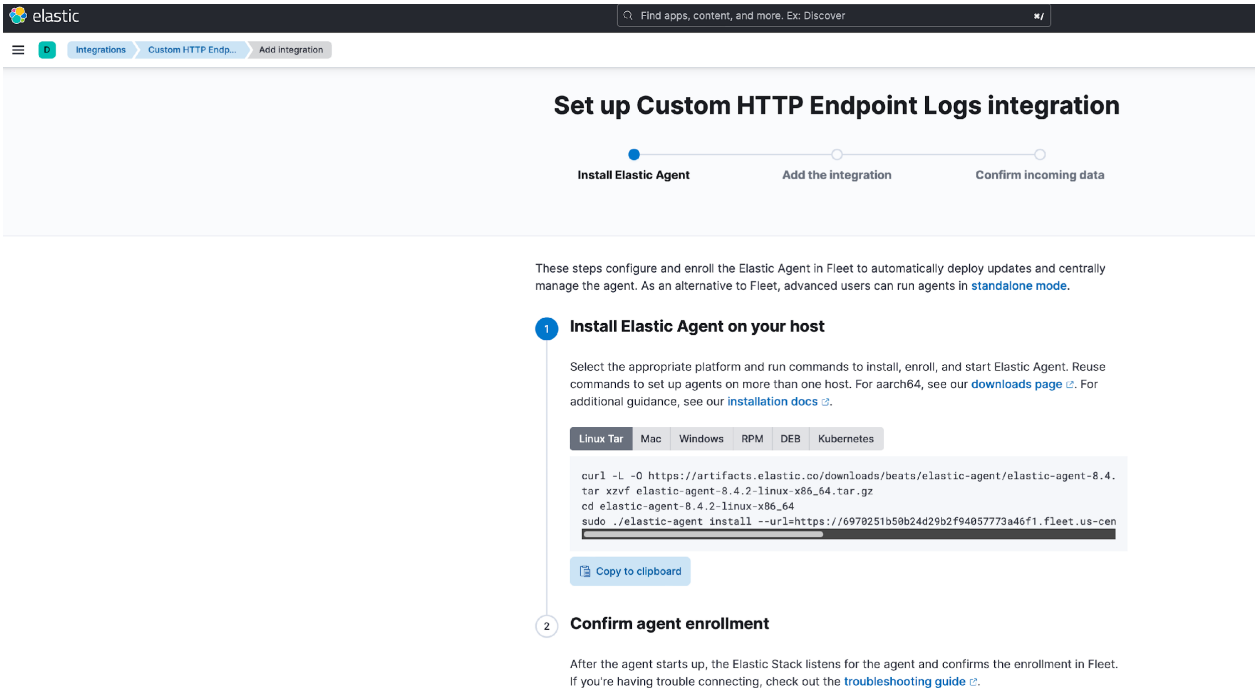
After a few seconds, the deployment should be ready. Click on “Continue”



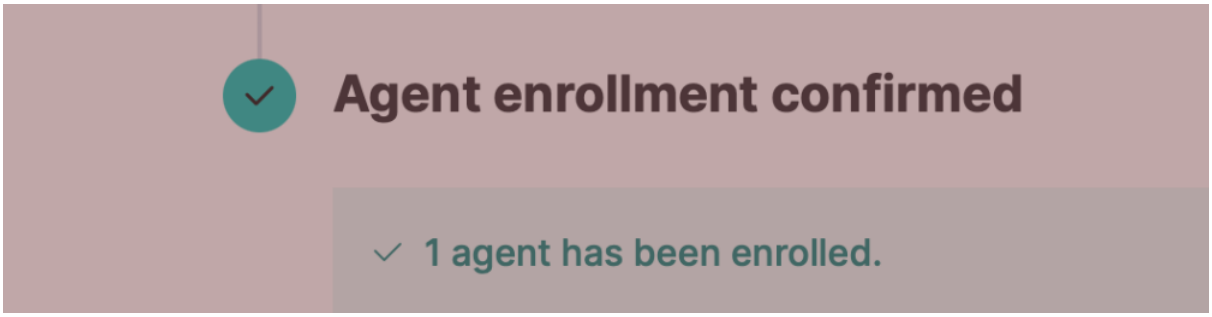
3.2 Install Agent, Add to Fleet




elastic
Find apps, content, and more. Ex: Discover
Integrations
Custom HTTP Endp...
Add integration
Ready to add your first integration?
1
2
3
Install Elastic Agent
Install agents on the hosts that you want to connect to Elastic.
Add the integration
Make a few selections to finalize how Elastic receives your data.
Confirm incoming data
Explore and analyze the incoming data.
Learn more about installing Elastic Agent
Go back
Install Elastic Agent


elastic
Find apps, content, and more. Ex: Discover
Integrations
Custom HTTP Endp...
Add integration
Set up Custom HTTP Endpoint Logs integration
Install Elastic Agent
Add the integration
Confirm incoming data
These steps configure and enroll the Elastic Agent in Fleet to automatically deploy updates and centrally manage the agent. As an alternative to Fleet, advanced users can run agents in standalone mode.
1
Install Elastic Agent on your host
Select the appropriate platform and run commands to install, enroll, and start Elastic Agent. Reuse commands to set up agents on more than one host. For aarch64, see our downloads page. For additional guidance, see our installation docs.
Linux Tar
Mac
Windows
RPM
DEB
Kubernetes
curl -L -O https://artifacts.elastic.co/downloads/beats/elastic-agent/elastic-agent-8.4.
tar xzvf elastic-agent-8.4.2-linux-x86_64.tar.gz
cd elastic-agent-8.4.2-linux-x86_64
sudo ./elastic-agent install --url=https://6970251b50b24d29b2f94057773a46f1.fleet.us-cen
Copy to clipboard
2
Confirm agent enrollment
After the agent starts up, the Elastic Stack listens for the agent and confirms the enrollment in Fleet. If you're having trouble connecting, check out the troubleshooting guide.

Choose the agent platform as per you choice for example a host running in AWS EC2. Once the agent is successfully installed, the custom agent enrollment will display “agent has been enrolled”



You will be able to see the agent listed under Fleet, under Elastic web console=>Management=>Fleet

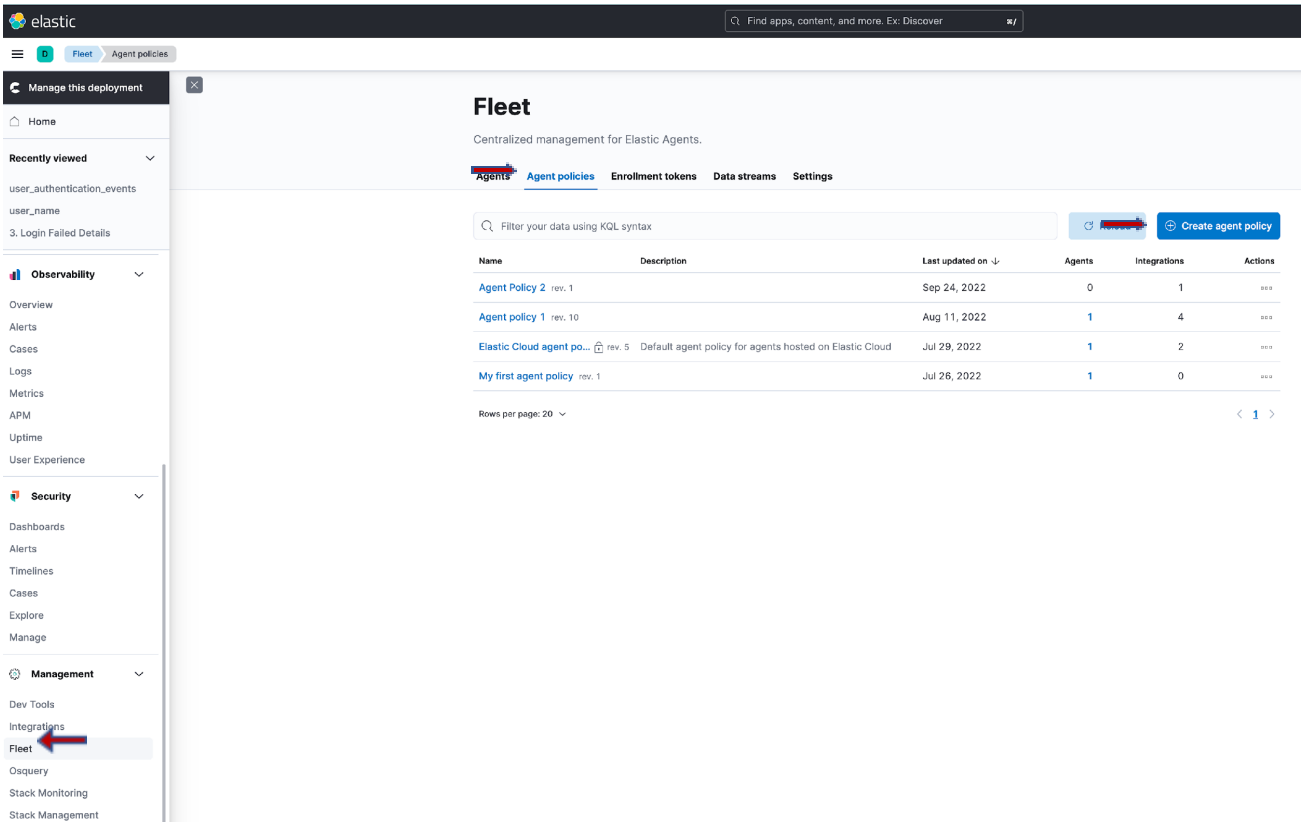
3.3 Create Agent Policy / Add agent

Agents are added to an agent policy and then integrations are attached to the policy. A policy is a collection of inputs and settings that defines the data to be collected by an Elastic Agent. Each Elastic Agent can only be enrolled in a single policy.

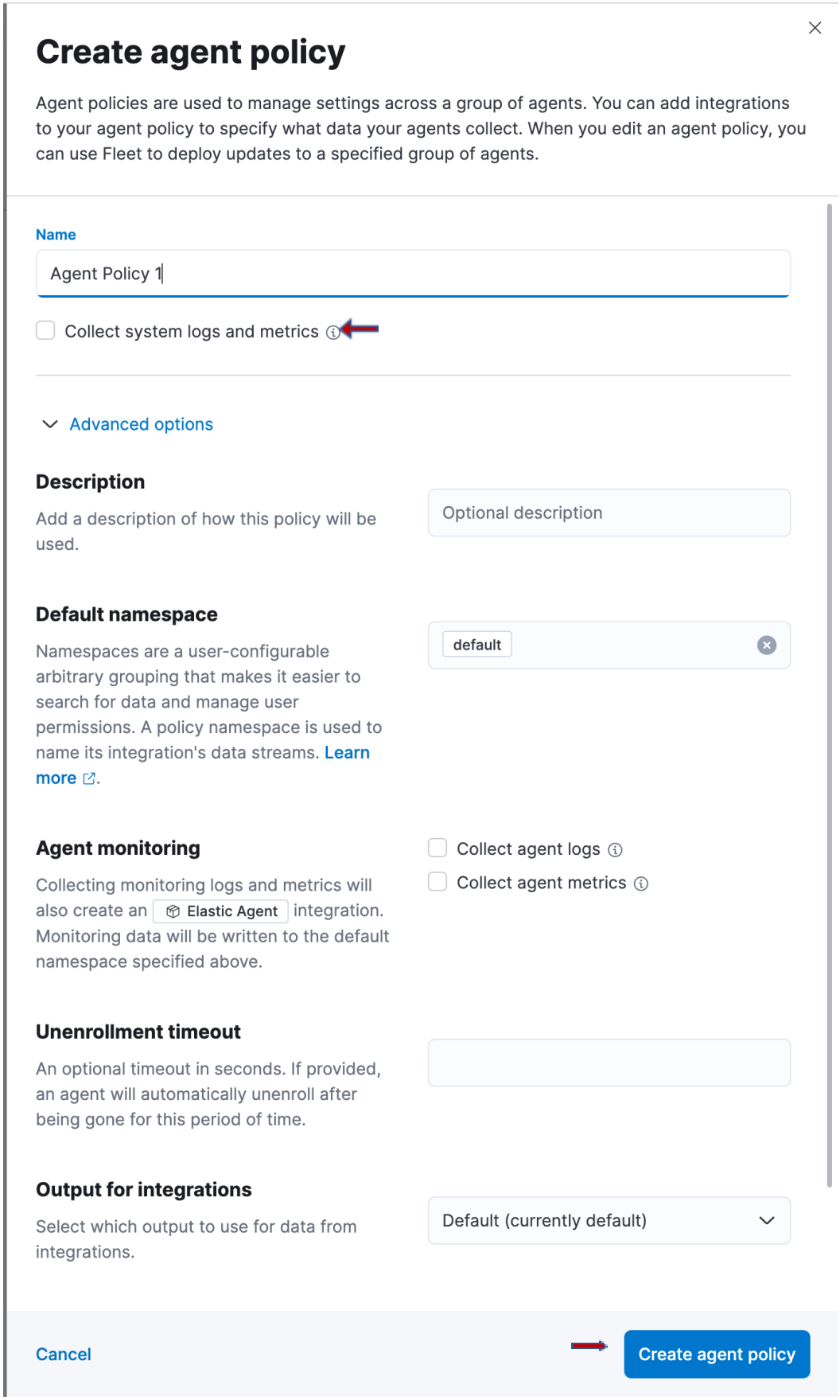
Within an Elastic Agent policy is a set of individual integration policies. These integration policies define the settings for each input type. The available settings in an integration depend on the version of the integration in use.

custom HTTP Endpoint Log integration is used for setting up a HTTP listener to post BI data events.
custom HTTPJSON input integration is used to ingest data from BI tenant events API endpoint[https://dataexport-public.byndid.com/v1/events?ordering=desc] to pull data.

Navigate to Fleet under Management, and choose “Agent Policies” Tab. Click on “create agent policy”

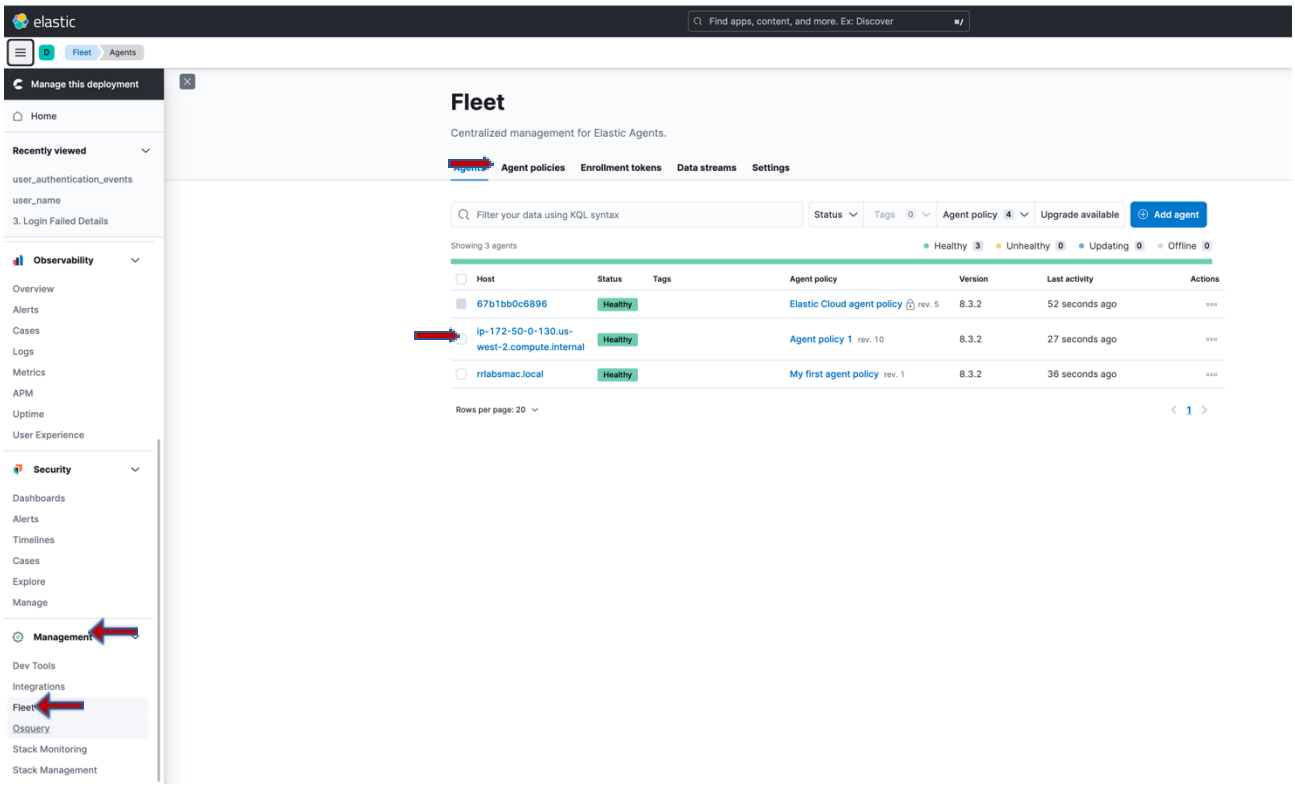


Provide a name for the agent policy, for example “Agent Policy 1”. Uncheck “Collect system logs and metrics”. Under “Advanced option”, uncheck “collect agent logs” and “collect agent metrics”. Click “Create agent policy”

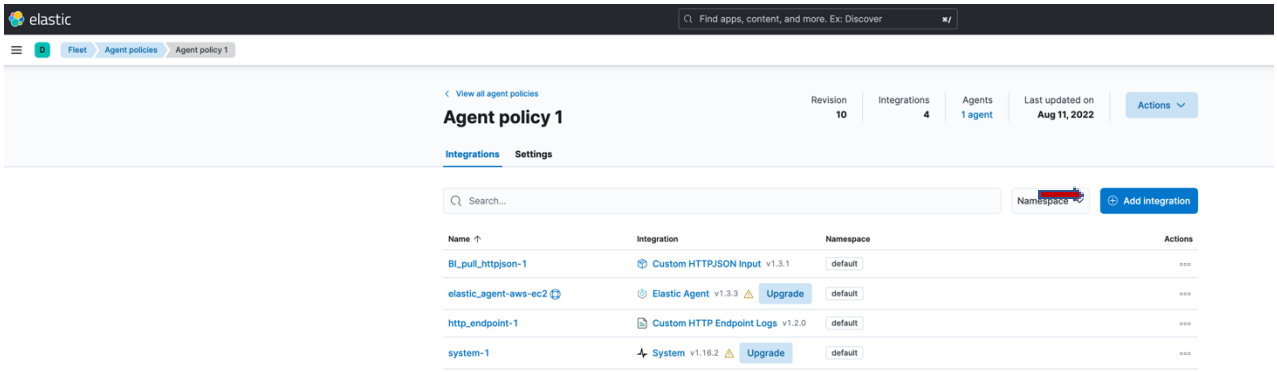

Create agent policy
Agent policies are used to manage settings across a group of agents. You can add integrations to your agent policy to specify what data your agents collect. When you edit an agent policy, you can use Fleet to deploy updates to a specified group of agents.
Name
Agent Policy 1
Collect system logs and metrics
Advanced options
Description
Add a description of how this policy will be used.
Optional description
Default namespace
Namespaces are a user-configurable arbitrary grouping that makes it easier to search for data and manage user permissions. A policy namespace is used to name its integration's data streams. Learn more.
default
Agent monitoring
Collecting monitoring logs and metrics will also create an Elastic Agent integration. Monitoring data will be written to the default namespace specified above.
Collect agent logs
Collect agent metrics
Unenrollment timeout
An optional timeout in seconds. If provided, an agent will automatically unenroll after being gone for this period of time.
Output for integrations
Select which output to use for data from integrations.
Default (currently default)
Cancel
Create agent policy

3.4 Add HTTP Logs Integration to agent policy [Push]

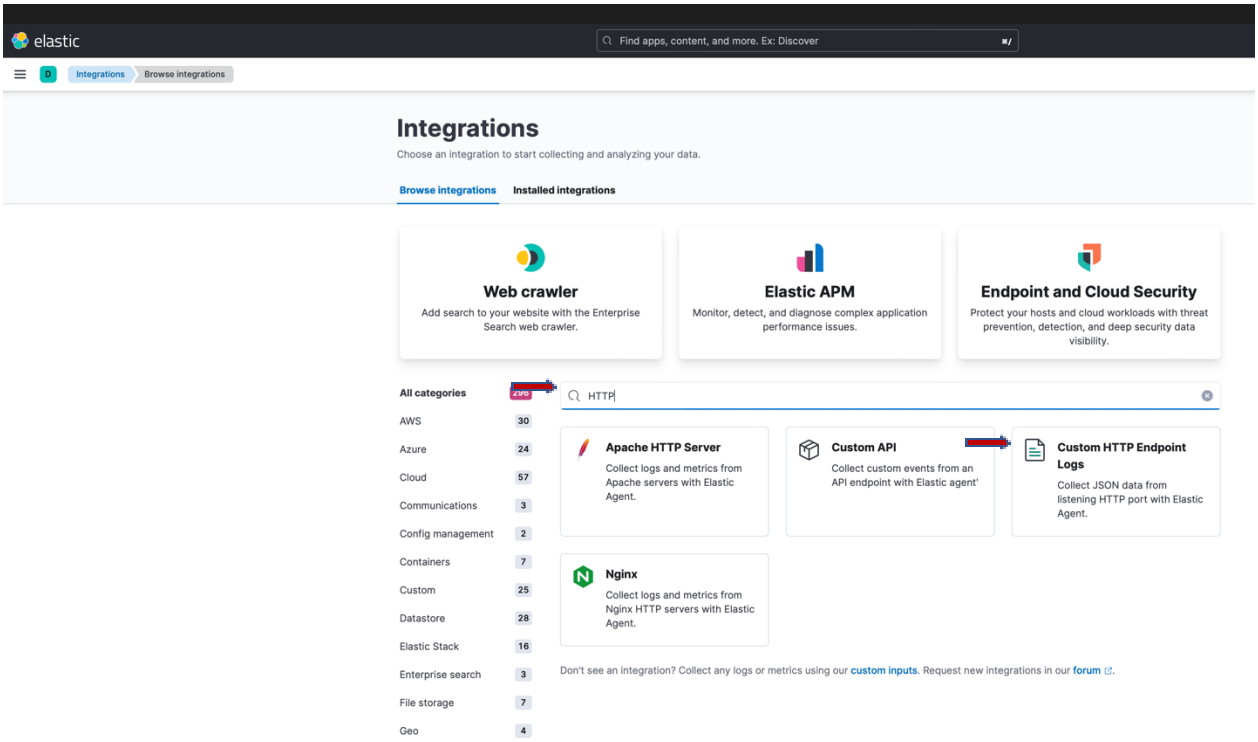
Navigate to Management=>Fleet. Click on “Agent policies”.



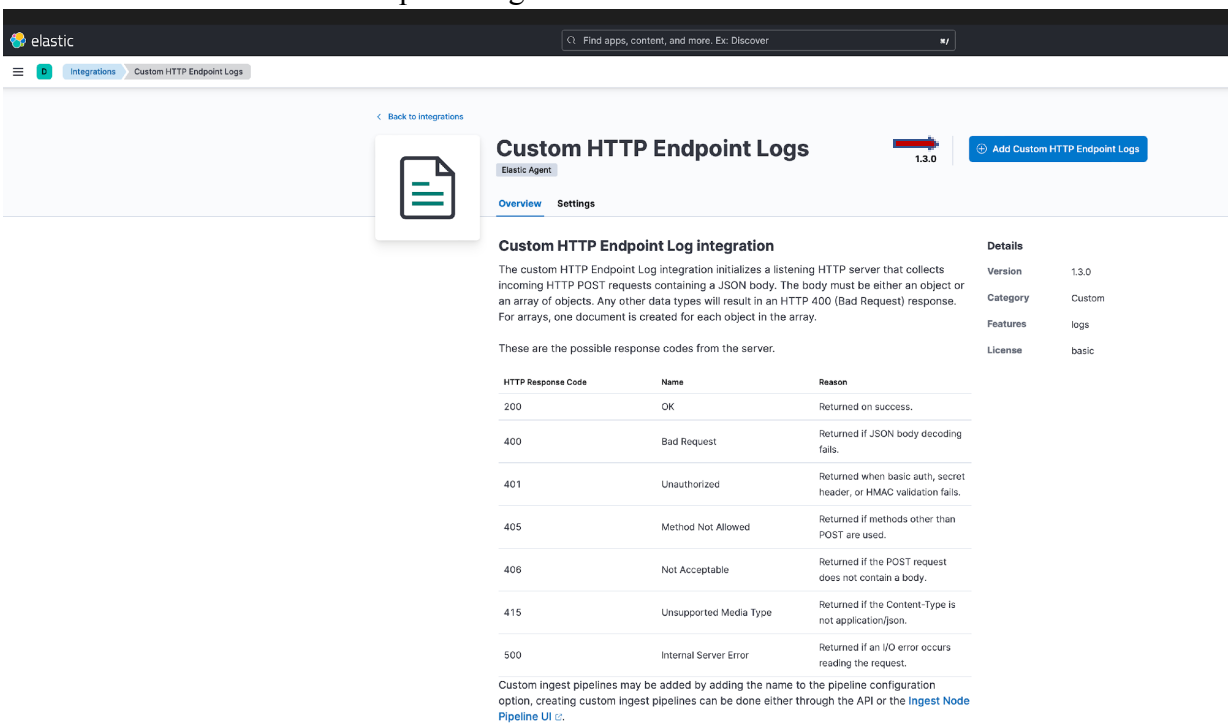
Click on “Add integrations”



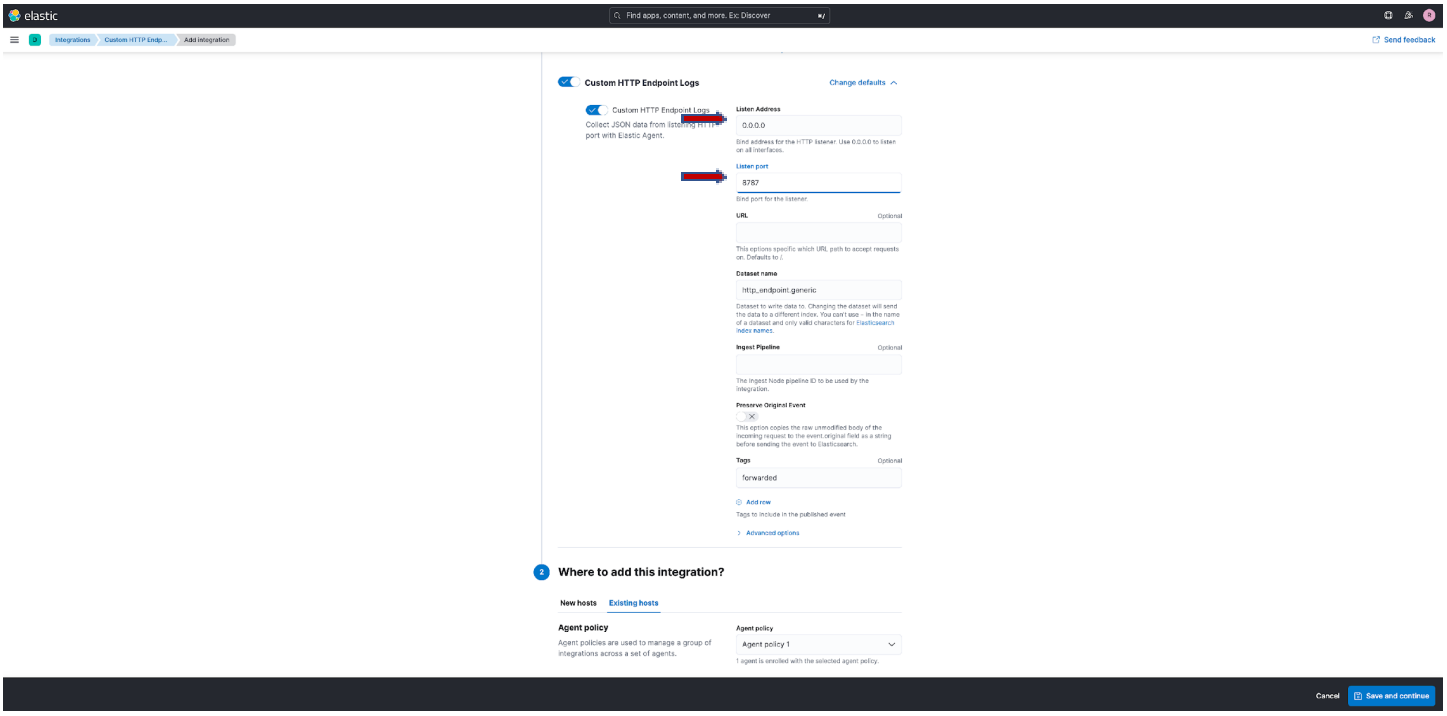
Type “HTTP” in the search box. In the search results, click “custom HTTP Endpoint Logs”



Click “Add Custom HTTP Endpoint Logs”

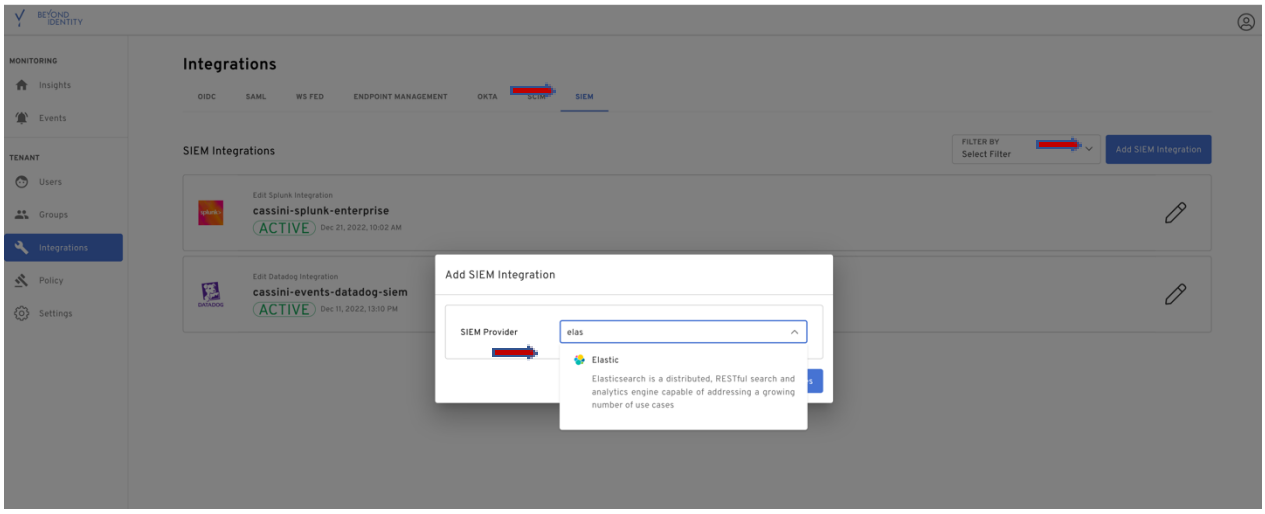


Enter “0.0.0.0” for the Listen Address and a value for the listen port, for example 8787. Please note this port must be opened to Beyond Identity to post events. Leave the other settings to defaults. Under “Where to add this integration?”, choose the agent policy created in the section above “Agent Policy 1”

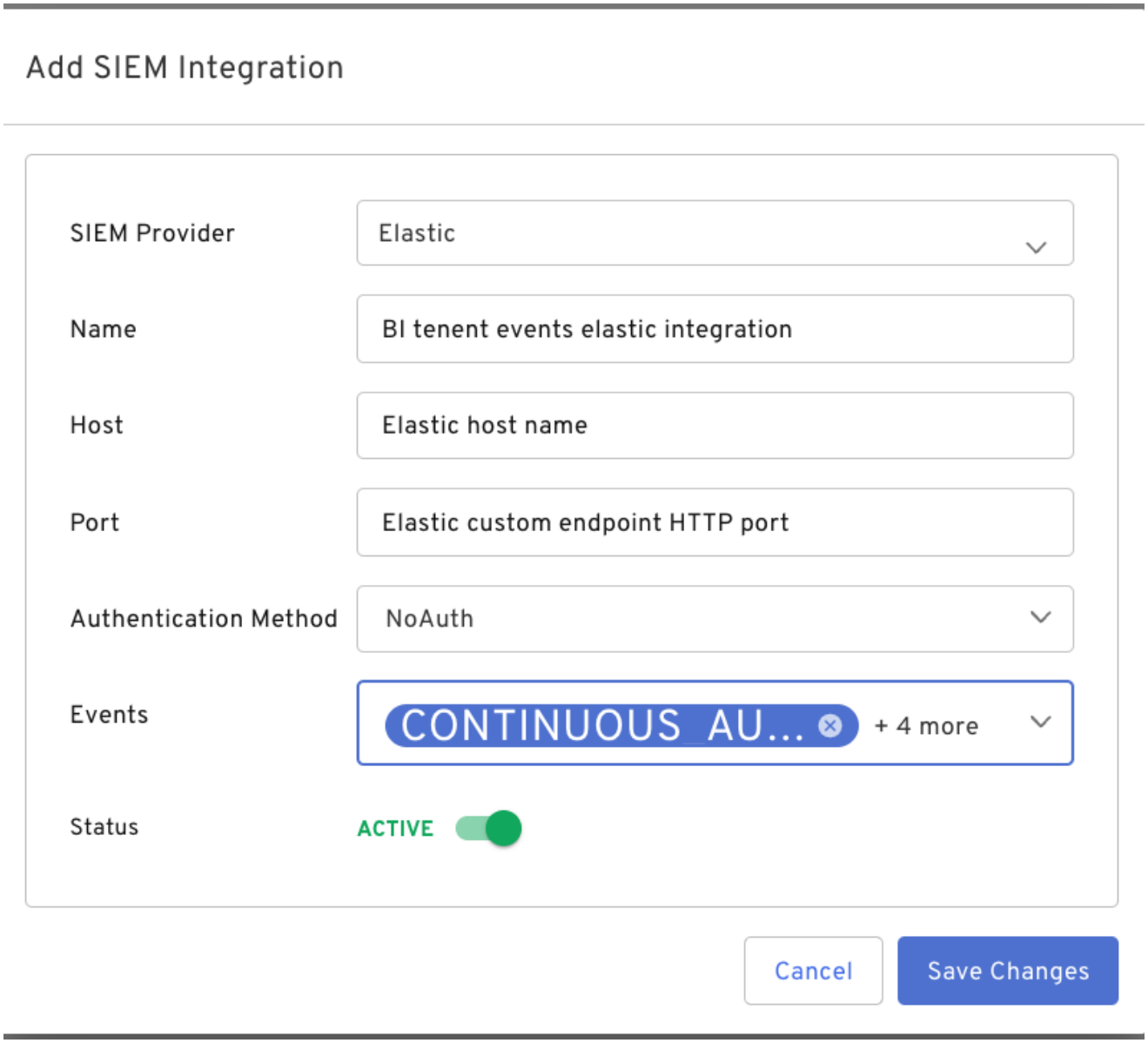


4 Beyond Identity Configuration

The configuration is done using the BI admin console. Access BI admin console through your SSO integration. Click on “Integrations” and click on “SIEM”. Click on “Add SIEM Integration”



Choose “Elastic” from the drop down.

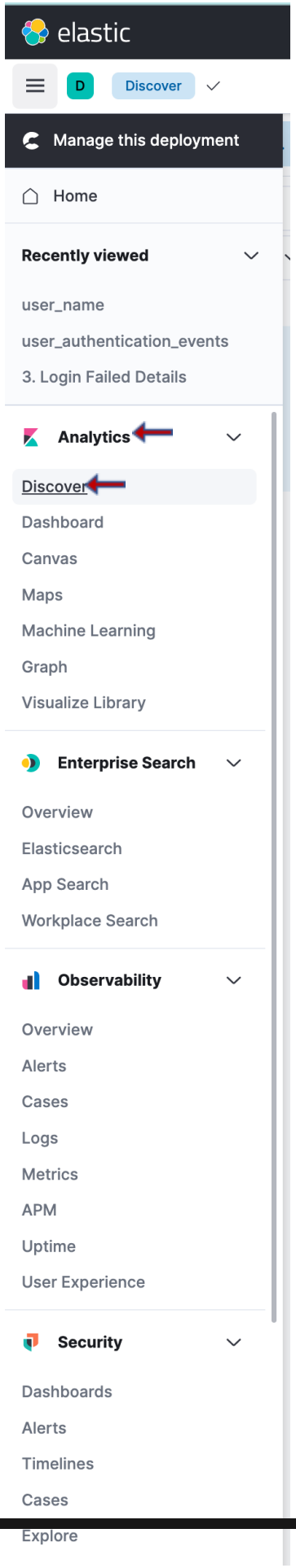


Provide a name for the configuration. From the events drop down, “select all” events or one the events you are interested in. Click on “Save Changes”

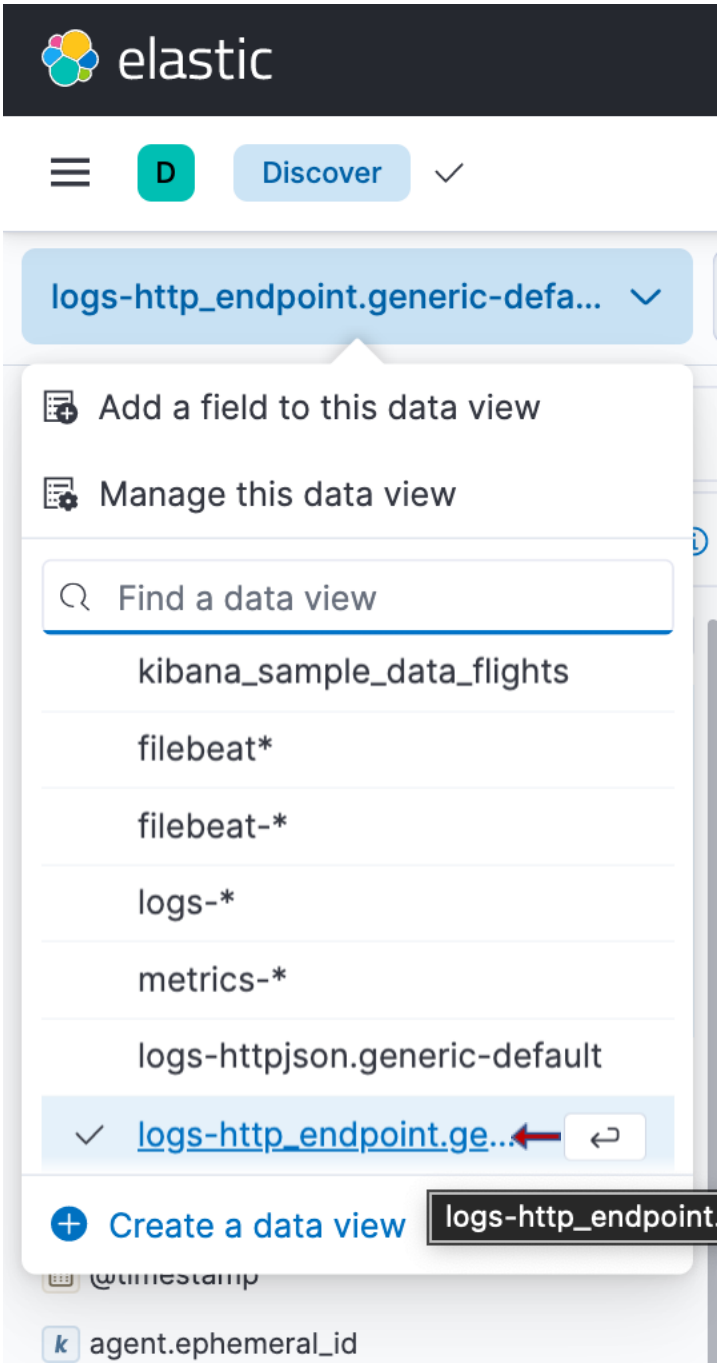
Once SIEM configuration is complete in BI admin console, you will be able to see the events in your Elastic. You can verify with a log search in Elastic, for example

5 Verification in Elastic

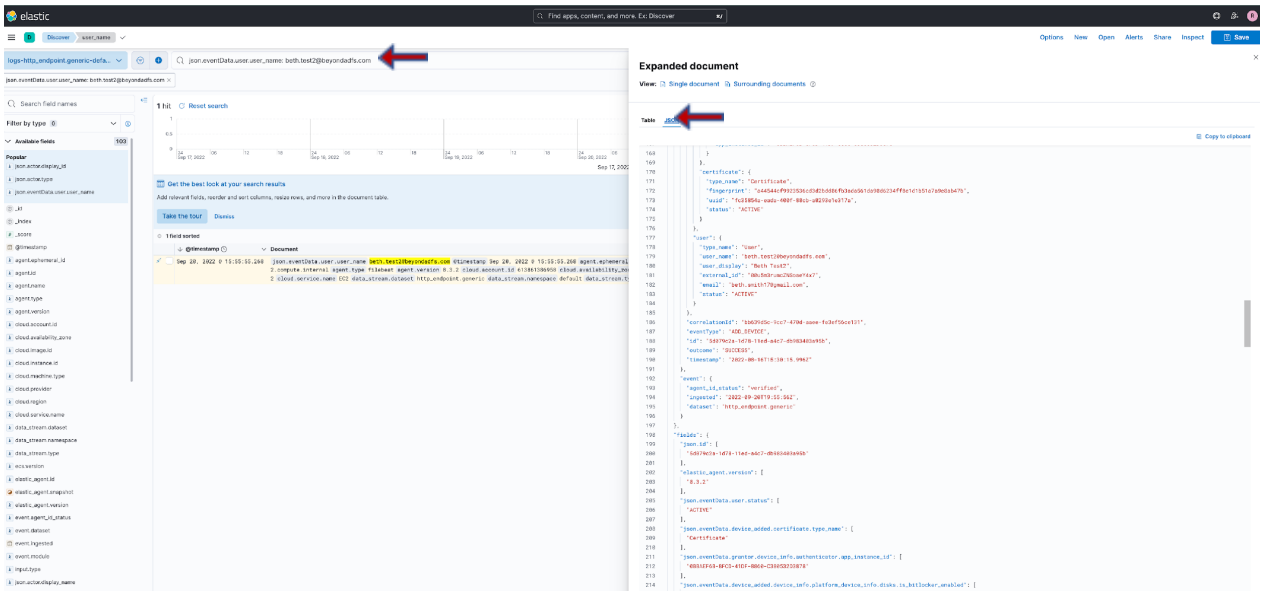
- Access your Elastic URL
- Select “Discover” under “Analytics”


elastic
D
Discover
Manage this deployment
Home
Recently viewed
user_name
user_authentication_events
3. Login Failed Details
Analytics
Discover
Dashboard
Canvas
Maps
Machine Learning
Graph
Visualize Library
Enterprise Search
Overview
Elasticsearch
App Search
Workplace Search
Observability
Overview
Alerts
Cases
Logs
Metrics
APM
Uptime
User Experience
Security
Dashboards
Alerts
Timelines
Cases
Explore

Choose logs-http_endpoint.generic-default



Enter “json.eventData.user.user_name: USER_WHO_AUTHENTICATED_USING_BI“ in the query box. Select the event in the results and click “JSON” on the right.

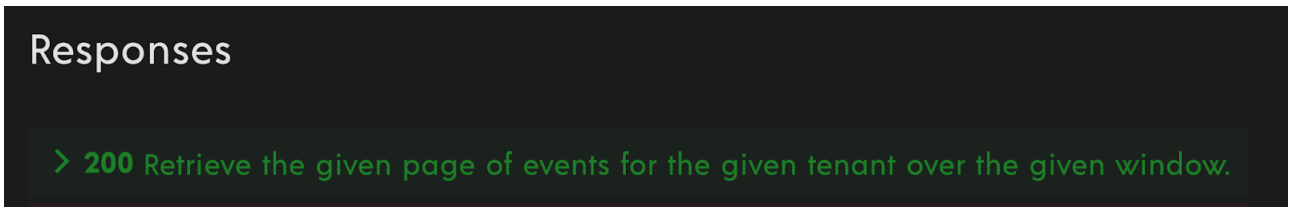


6 Appendix B

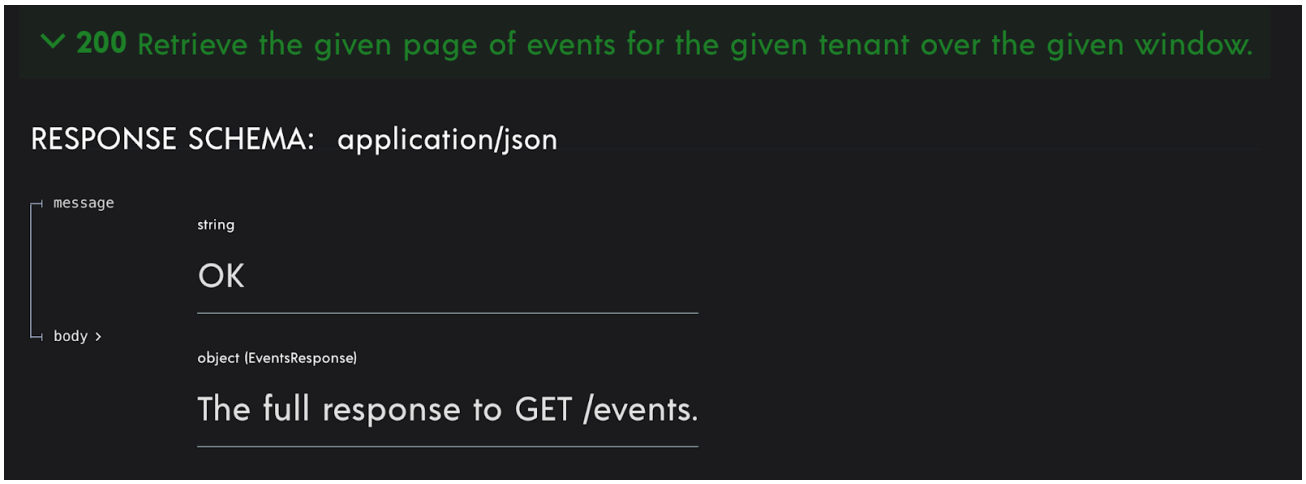
How to get event types?

Click on https://developer.beyondidentity.com/api/v0#tag/Events/operation/getEvents

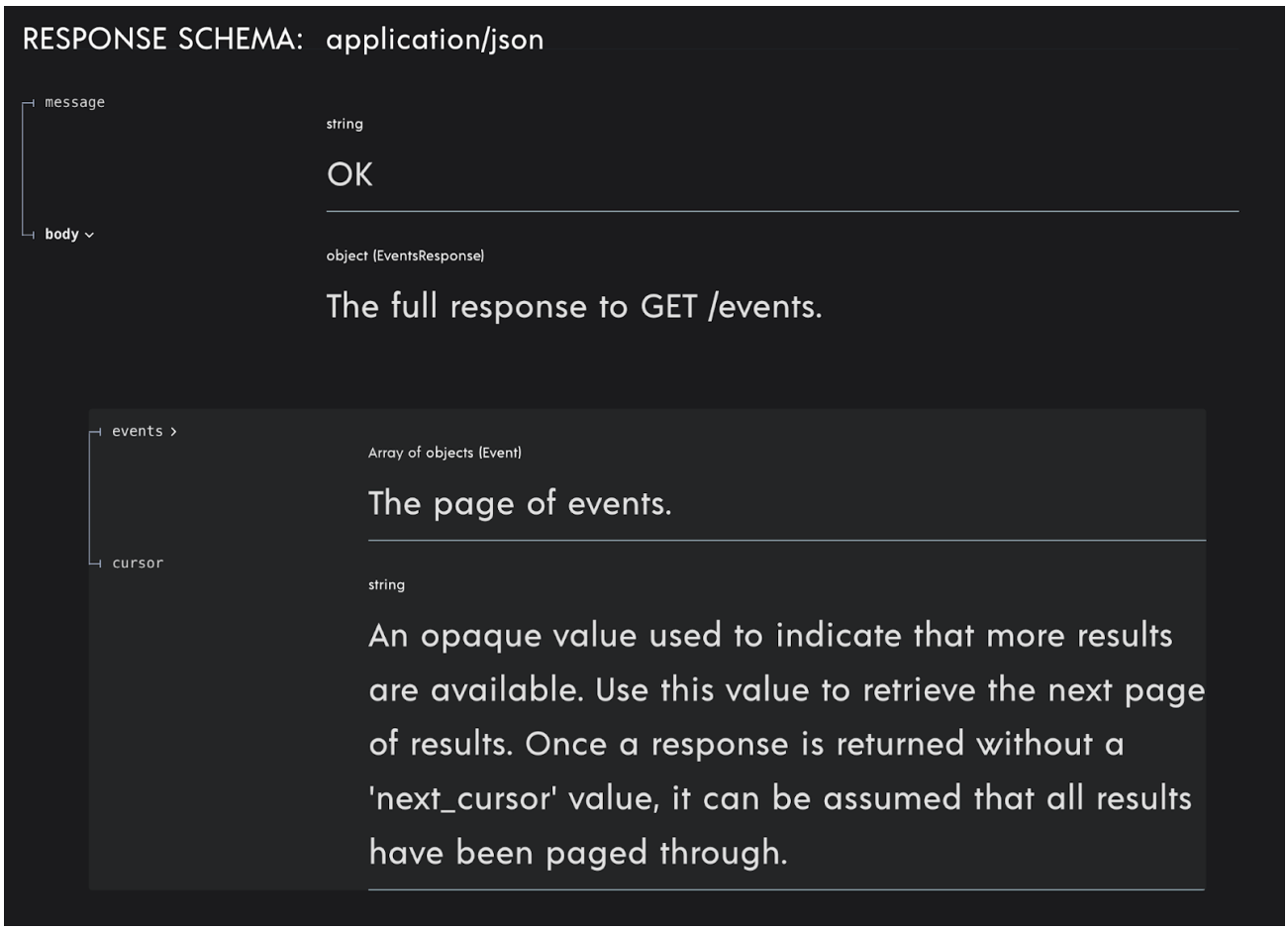
Scroll down



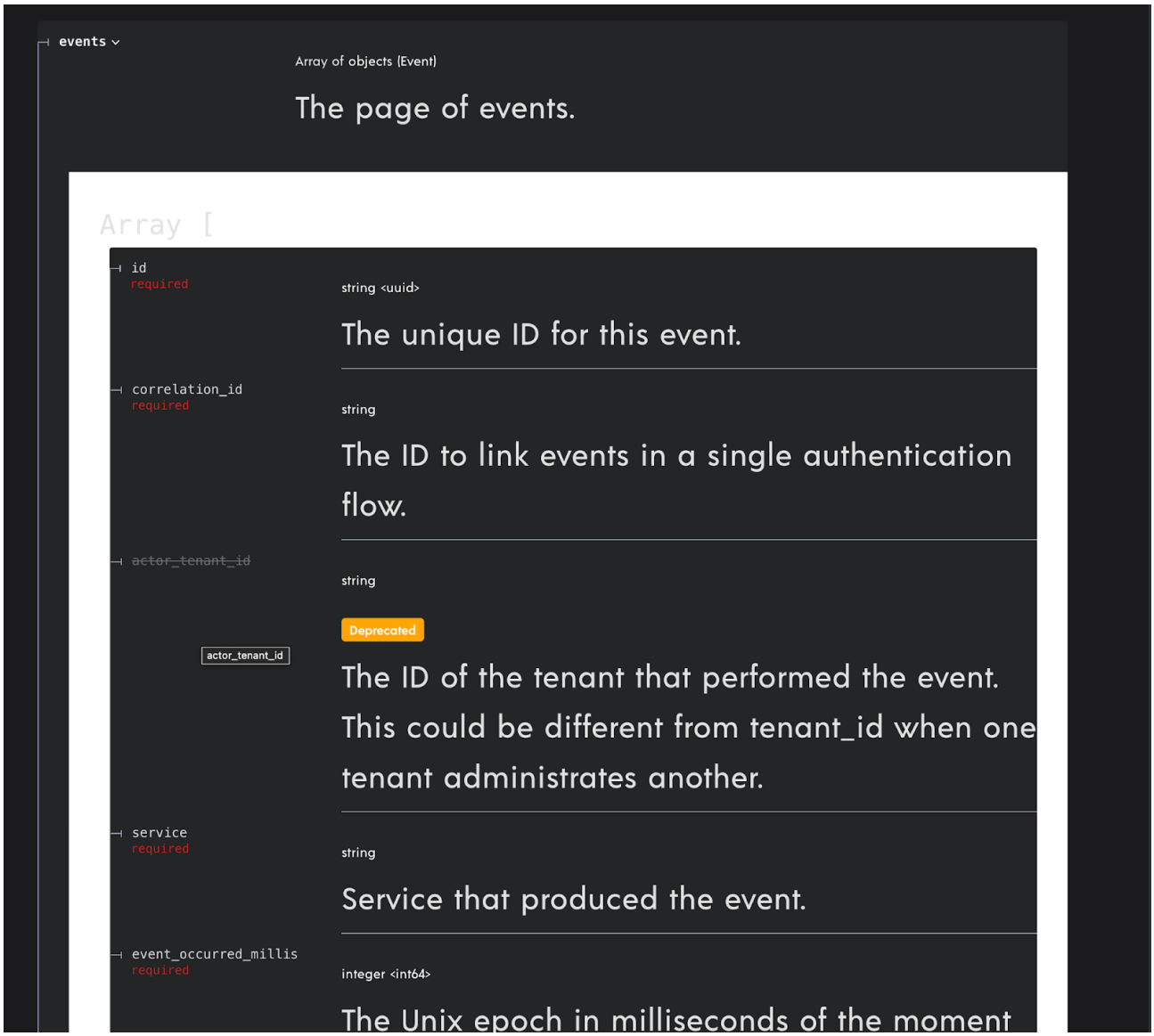
Click on arrow next to 200



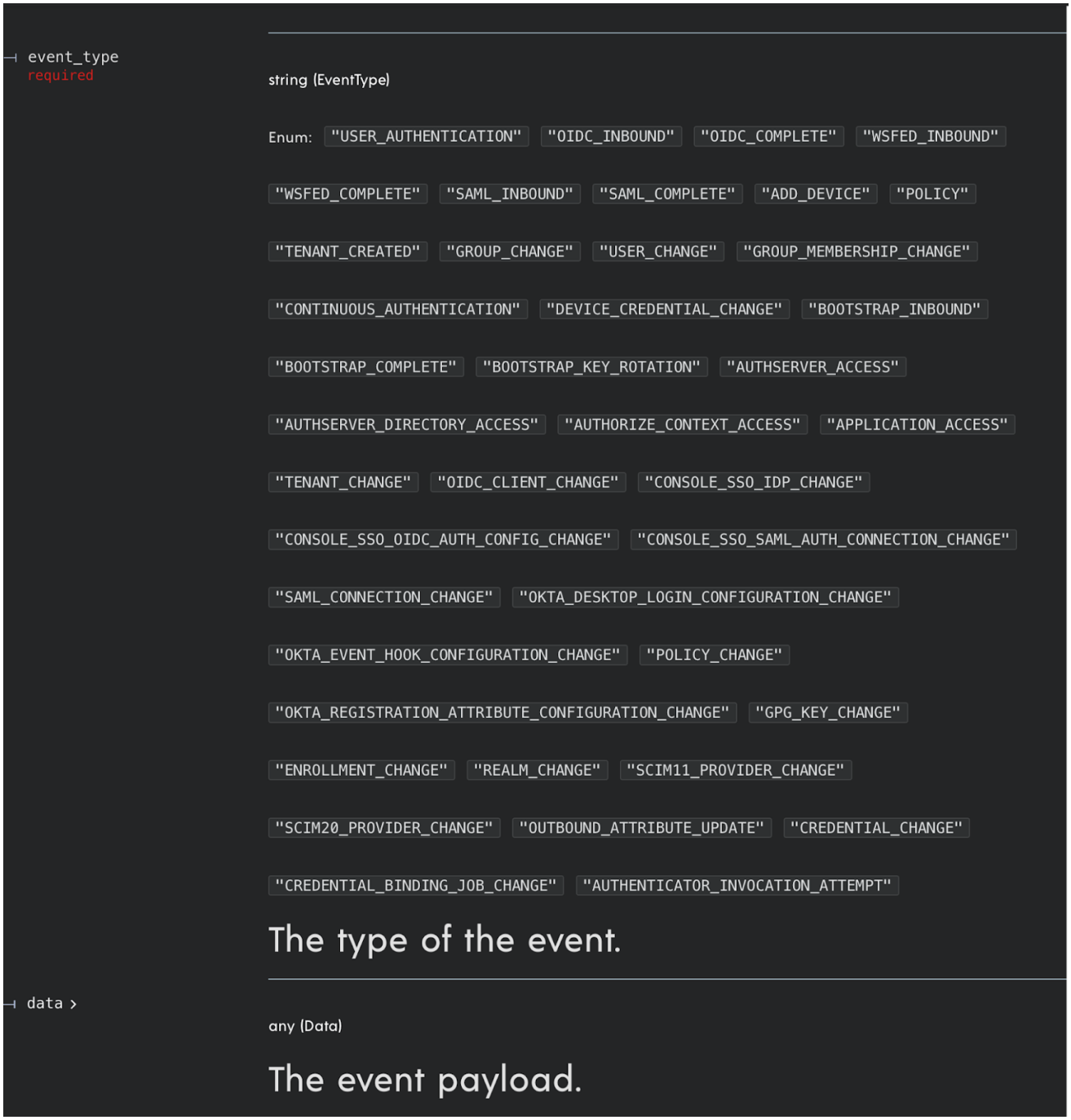
Click on body



Click on events


events
Array of objects (Event)
The page of events.
Array [
id
required
string <uuid>
The unique ID for this event.
correlation_id
required
string
The ID to link events in a single authentication flow.
actor_tenant_id
string
Deprecated
actor_tenant_id
The ID of the tenant that performed the event. This could be different from tenant_id when one tenant administrates another.
service
required
string
Service that produced the event.
event_occurred_millis
required
integer <int64>
The Unix epoch in milliseconds of the moment

Scroll down



event_type lists all the events