PREFECT

Prefect may have access to four types of customer data, depending on the customer's use of Prefect: (i) non-sensitive workflow metadata, (ii) user data, (iii) authentication data (optional feature) and (iv) log data (optional feature), each described in further detail below.

1. Non-sensitive workflow data.

Because of Prefect's patented [hybrid execution model](#), all customer data, storage, and execution remains within customer's environment (on-prem or cloud). To facilitate this hybrid execution, Prefect Deployments, and Workers require the following information:
- "where" to execute (Docker, ECS, Process, etc.) the code.
- "what" code to execute (the executable Python code) as well as any potential inputs required for the code to execute. This could be code that is stored locally on a file system, in a Github repo, or in a docker container.
- "when" to execute the code (through various scheduling rule sets).
- conditional execution variables, such as automations, event-driven workflows, or webhooks.

Note that Prefect does not have access to such data/code, but instead only an indication of where such data/code is located. The customer retains full control of the permissions and access necessary to retrieve and execute the data/code.

2. User data.

Prefect requires customer user names and email addresses to enable login, but not passwords (because of magic link authentication).

3. Authentication data.

Customer can elect to configure Kubernetes or container-based worker solutions. To facilitate loading these images during execution, the requisite connection strings, URLs, or credentials can be saved in Prefect Cloud. Use of these features would give Prefect access to such data when accessible via public internet. Storing credentials and connection strings is optional and risk can be mitigated through the use of existing key vaults within customer's cloud provider. Additionally, IRSA / workload Identity is commonly used to manage authentication at the cloud provider level.

4. Log data.

Finally, Prefect offers an integrated logger, which can be turned on and off at customer's discretion. By default, the logger logs the metadata of the flow execution (i.e., scheduled -> pending -> running, task creations and completions).