Create a Field Extraction Rule

You can create a field extraction rule of your own from scratch, or depending on your data source, you can use one of our templates that we provide in the drop-down menu.

You need the Manage field extraction rules role capability to create a field extraction rule. Fields specified in field extraction rules are automatically added and enabled in your Fields table schema.

Limitations

Ingest Time FERs have the following limitations.

- There is a limit of 50 Ingest Time rules and 200 fields. Fields created as log metadata and from Ingest Time rules share the same quota of 200 fields. You can manage your fields on the Fields page.
- Ingest Time rule expressions are limited to a maximum of 16k (16,384) characters.
- Ingest Time rules can extract up to a maximum of 16k (16,384) characters for each field.
- The cumulative size of all fields extracted by a rule for a message/event is limited to 64kb.
- Ingest Time rules only apply to data moving forward. If you want to parse data ingested before the creation of your Ingest Time FER, you can either parse your data in your query, or create Scheduled Views to extract fields for your historical data.

Creating a new Field Extraction Rule

To create a Field Extraction Rule:

1. Go to Manage Data > Logs > Field Extraction Rules.
2. Click the + Add button on the top right of the table.
3. The following form appears:
4. Enter the following options:

   - **Rule Name.** Type a name that makes it easy to identify the rule.

   - **Applied At.** There are two types available, Ingest Time and Run Time. The main differences are Run Time only supports JSON data and the time that Sumo parses the fields. The following is an overview of the differences:

     - **Ingest Time**

       - Parsing support - any data format, requires manually written parser expressions.

       - **Rule limit - There is a limit of 50 Field Extraction Rules and 200 fields. This includes the default fields defined by Sumo Logic (about 16). The 200-field limit is per account, and deleting rules does not**
create more space.

- **Time** - At the time of ingestion, only applies to data moving forward. If you want to parse data ingested before the creation of your FER, you can either parse your data in your query, or create Scheduled Views to extract fields for your historical data.

- **Run Time**
  - Parsing support - JSON, automatically
  - Rule limit - none
  - **Time** - During a search when using **Auto Parse Mode** from **Dynamic Parsing**.

  **Scope.** Select either **All Data** or **Specific Data**. When specifying data the options for the scope differ depending on when the rule is applied.

  - For an **Ingest Time** rule, type a **keyword search expression** that points to the subset of logs you'd like to parse. Think of the Scope as the first portion of an ad hoc search, before the first pipe ( | ). You'll use the Scope to run a search against the rule. Custom metadata fields are not supported here, they have not been indexed to your data yet at this point in collection.

  - For a **Run Time** rule, define the scope of your JSON data. You can define your JSON data source as a **Partition Name**(index), sourceCategory, Host Name, Collector Name, or any other **metadata** that describes your JSON data. Think of the Scope as the first portion of an ad hoc search, before the first pipe ( | ). You'll use the Scope to run a search against the rule. You cannot use keywords like “info” or “error” in your scope.

    Always set up JSON auto extraction (Run Time field extraction) on a specific Partition name (recommended) or a particular Source. Failing to do so might cause the auto parsing logic to run on data sources where it is not applicable and will add additional overhead that might deteriorate the performance of your queries.

    **Best Practices**
    1. If you are not using Partitions we recommend using **metadata** fields like _sourceCategory, _sourceHost or _collector to define the scope.
    2. We recommend creating a separate Partition for your JSON dataset and use that Partition as the scope for run time field extraction. For example, let's say you have AWS Cloudtrail logs, and they are stored in _view=cloudtrail Partition in Sumo. You can create a Run Time FER with the scope _view=cloudtrail. Creating a separate Partition and using it as scope for a run time field extraction ensures that auto parsing logic only applies to necessary Partitions.

  **Parsed template** (Optional for Ingest Time rules).
  - Click the drop down under **Parsed template** to see the available templates.
  - Choose a template and click **Use Template**. The template is applied to the Parse Expression.

  **Parse Expression.** (Applicable to Ingest Time rules)
  - Type a valid parse expression with **supported parse and search operators**. Because fields are associated with the Rule Name, you can parse one particular field into as many rules as you'd like.
For example, to parse a single field, you could use a definition similar to this:

```plaintext
parse "message count = \*," as msg_count
```

To parse multiple fields, you could use a definition similar to this:

```plaintext
parse "[hostId=\*] [module=\*] [localUserName=\*] [logger=\*] [thread=\*]" as hostId, module,
localUserName, logger, thread
```

5. **Extracted Fields** (applicable to Ingest Time rules) shows the field names the rule will parse. Any fields that do not exist in the Field table schema are shown with the text New highlighted in green. New fields are automatically created in the table schema when you save the rule. You can view and manage the field table schema on the Fields page.

6. Click **Save** to create the rule.

---

**Best practices for designing Rules**

Include the most accurate keywords to identify the subset of data from which you want to extract data. Lock down the scope as tightly as possible to make sure it's extracting just the data you want, nothing more. Using a broader scope means that Sumo Logic will inspect more data for the fields you'd like to parse, which may mean that fields are extracted when you don't actually need them.

Create multiple, specific rules. Instead of constructing complicated rules, create multiple rules with basic scope, then search on more than one (rules are additive). The OR and AND commands are supported, just as in any search. For example, you could use one rule to parse Apache log response codes, and then use another rule to parse response time. When used together, you can get all of the information you may need.

Don't extract fields you don't need. Extract the minimum number of fields that should all be present in logs. Every field you include in the scope shows up in every search, so including extra fields means you'll see more results than you may need. It's better to create more rules that extract the fields that are most commonly used. First, look at common data sources and see what's most frequently extracted. Then, think about what you most frequently parse from those sources, then create rules to automatically extract those fields.

Create multiple parse nodrop statements in an FER for a field name to match distinct log patterns. The different parse statements will effectively function like an OR statement since only one will match the log message and return the field value.

Test the scope before creating the rule. Make sure that you can extract fields from all messages you need to be returned in search results. Test them by running a potential rule as a search.

Make sure all fields appear in the Scope you define. When Field Extraction is applied to data, all fields must be present to have any fields indexed; even if one field isn't found in a message, that message is dropped from the results. In other words, it's all or nothing. For multiple sets of fields that are somewhat independent, make two rules.

Re-use field names in multiple FERs if scope is distinct and separate and not matching same messages. To save space and allow for more FERs within your 200 field limit, you can re-use the field names as long as they are used in non-overlapping FERs.

Avoid targeting the same field name in the same message with multiple FERs. When more than one FER targets
the same message with the same field name, one of the rules will NOT apply. The rule applied to the specific field name is randomly selected. Don't use the same field names in multiple FERs that target the same messages.

**Supported parsing and search operators**

The following operators can be used as part of the **Parse Expression** in an Ingest Time Field Extraction Rule. The **multi** and **auto** options are not supported in FERs.

- parse regex
- parse anchor
- parse nodrop
- csv
- fields
- json
- keyvalue
- num