### Loggers

#### Overview

Loggers are used to collect data from data sources. The method by which a logger collects data may vary according to the type defined in the logger model. The logger model determines how data will be collected, and the logger performs the role of collecting data according to the method defined in the logger model.



Data Sources

Most data sources exist as network hosts external to Logpresso Sonar. A data source can be an application server providing services such as web applications, a file server, a database, or a network security device like a firewall. It may also be a service accessible via a REST API. PCAP devices can also serve as data sources.

The Logpresso Sonar server itself can also be a data source. There may be cases where it is necessary to collect event context data stored in memory for event correlation analysis, logs from Logpresso Sonar or Sentries, or other data derived from data collected from external data sources.

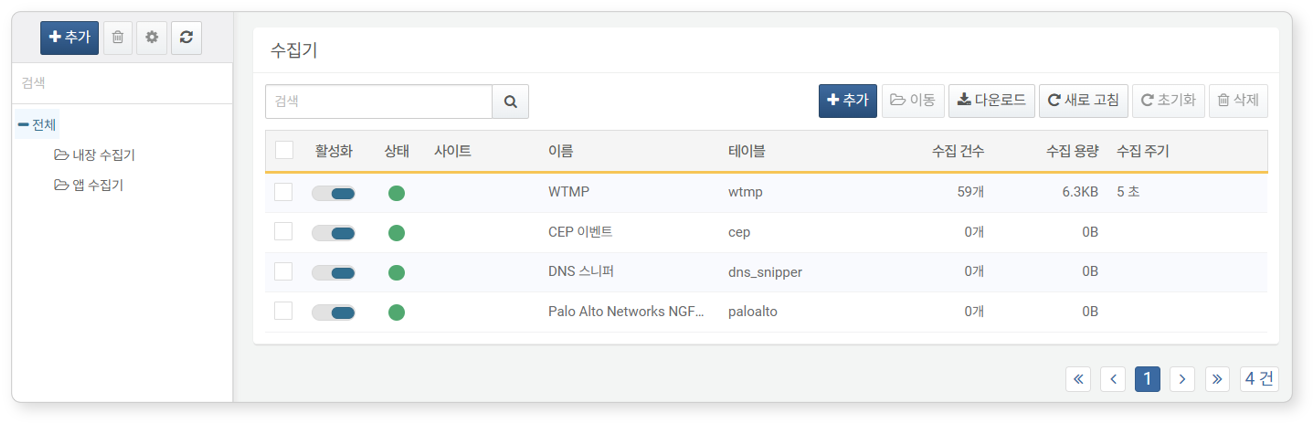
Classification by Collection Method

Loggers are classified into passive and active types based on how they collect data from data sources.

* **Passive Method**: Logpresso Sonar opens a data reception port, and the logger receives and processes the data sent by the data source. In the illustration above, loggers that collect data using protocols such as Syslog, SNMP Trap, NetFlow, and sFlow operate in a passive manner.
* **Active Method**: Logpresso Sonar directly accesses the data source at regular intervals or according to a specified schedule to collect data. In the illustration above, loggers that access data sources using protocols such as FTP, SFTP, SSH, HTTP, and JDBC operate in an active manner. Loggers that collect data via a REST API provided by a service also fall under the active method.

#### Viewing/Search Logger List

You can view or search the logger list under **Collection > Loggers**. The initial state of Logpresso Sonar does not provide any loggers, so it will only display an empty list. The following illustration shows an example of the logger list for the **All** group. (The default group, **All**, shows loggers belonging to all logger groups as well as those that do not belong to any group.)



* **Activate**: Toggle button for activating the logger.
* **Status**: Execution status of the logger (green: normal, gray: disabled, orange: collection failure). The logger may not operate if the node running the logger or the Sentry has not yet synchronized settings, or due to reasons such as disk exhaustion.
* **Site**: Site information for the data source.
* **Name**: The name used to identify the logger.
* **Table**: The name of the table where collected data is stored.
* **Collection Count**: The number of collected records.
* **Collection Capacity**: The capacity of collected records (default unit: bytes). This value may differ from the disk usage shown by the table.
* **Collection Period**: The data collection period or schedule for active loggers. Passive loggers do not have a collection period.

To find a specific logger in the logger list, use the search tool in the toolbar. The search tool will find and display loggers that contain the word entered in the **Name** field. The logger search tool is case-insensitive.

#### Logger Groups

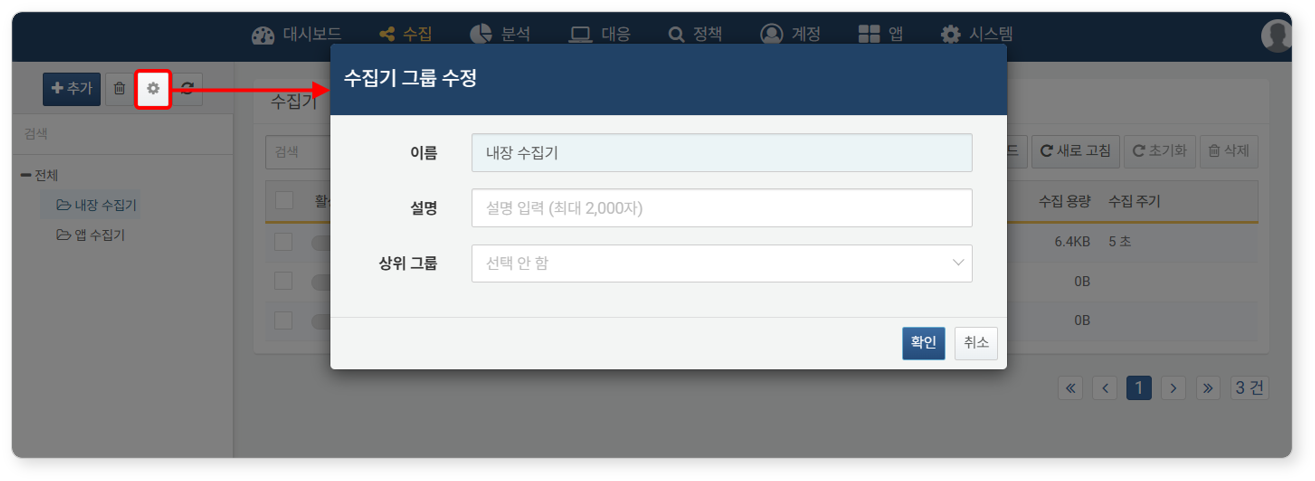
To manage loggers by grouping them, use logger groups.

To add a logger group, click **Add** above the logger group list in **Collection > Loggers**, then enter the **Name** and **Description** of the logger group, select the **Parent Group**, and click **OK**.

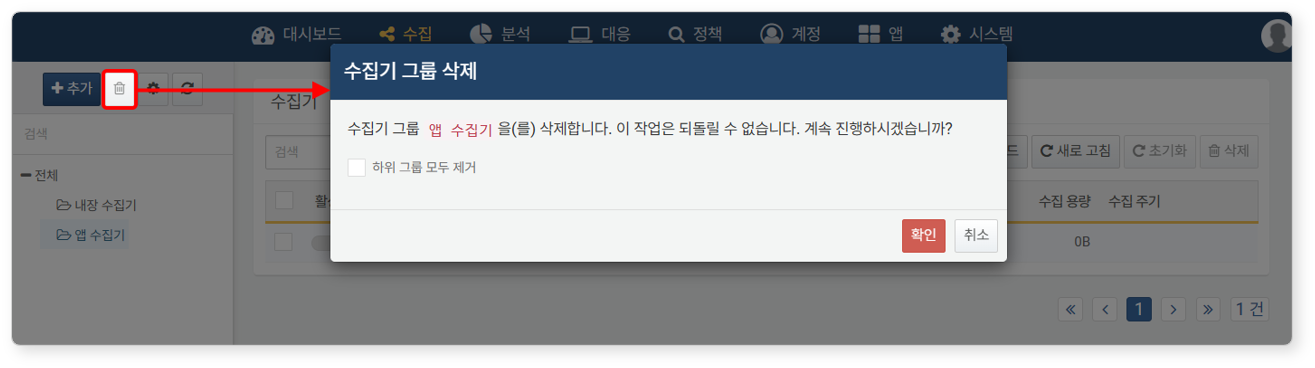


* **Name**: The name to identify the logger group.
* **Description**: A description of the logger group.
* **Parent Group**: The parent group of the logger group to be added (default: none selected).

To modify a logger group, select the logger group to be modified from the logger group list, then click the gear button.



To delete a logger group, select the logger group to be deleted from the logger group list, then click the trash button. To delete all groups under the logger group, select **Remove All Subgroups**.



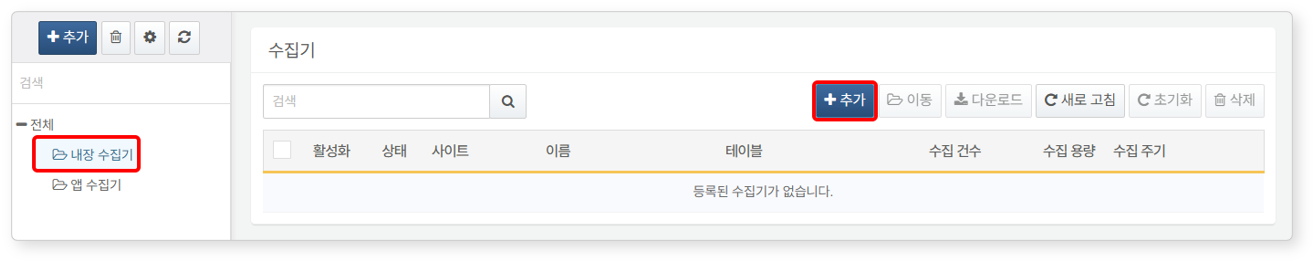
If the logger group to be deleted contains loggers, only the belonging groups will be removed, and the loggers will be left without a belonging group.

#### Adding Loggers

Unlike parsers, log schemas, and logger models, loggers must be created by the administrator to match the data sources to be collected. To add a logger:

Select the logger group to which you want to add the logger in **Collection > Loggers**. If you do not select a logger group, the logger will be created without a belonging group.

Click **Add** in the toolbar.



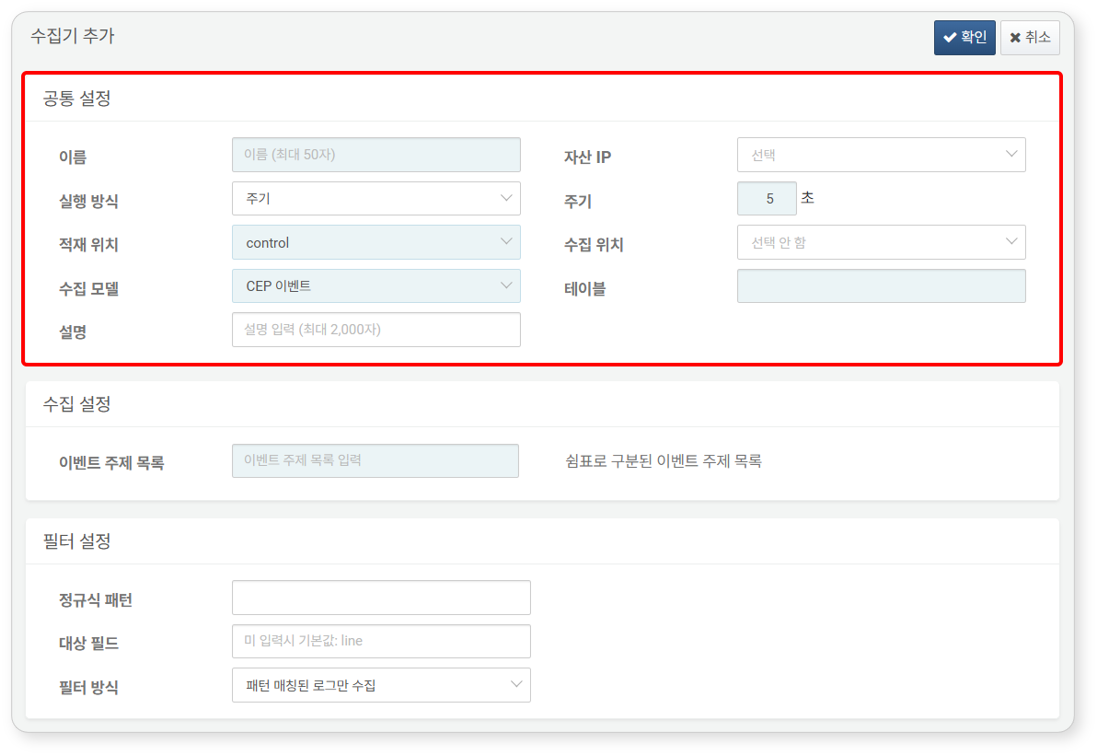
In the **Add Logger** screen, enter or select the necessary values for [Common Settings](https://docs.logpresso.comnull), [Collection Settings](https://docs.logpresso.comnull), and [Filter Settings](https://docs.logpresso.comnull), then click **OK**. The settings with a sky-blue background are required fields.

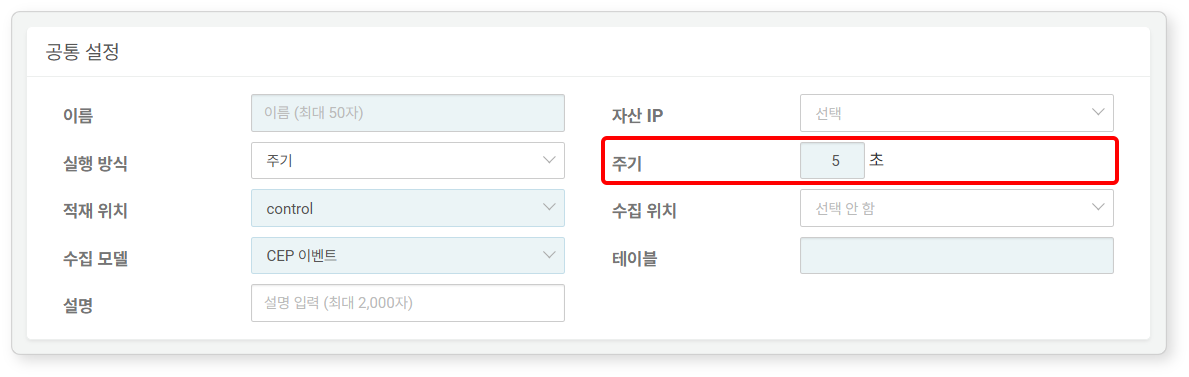
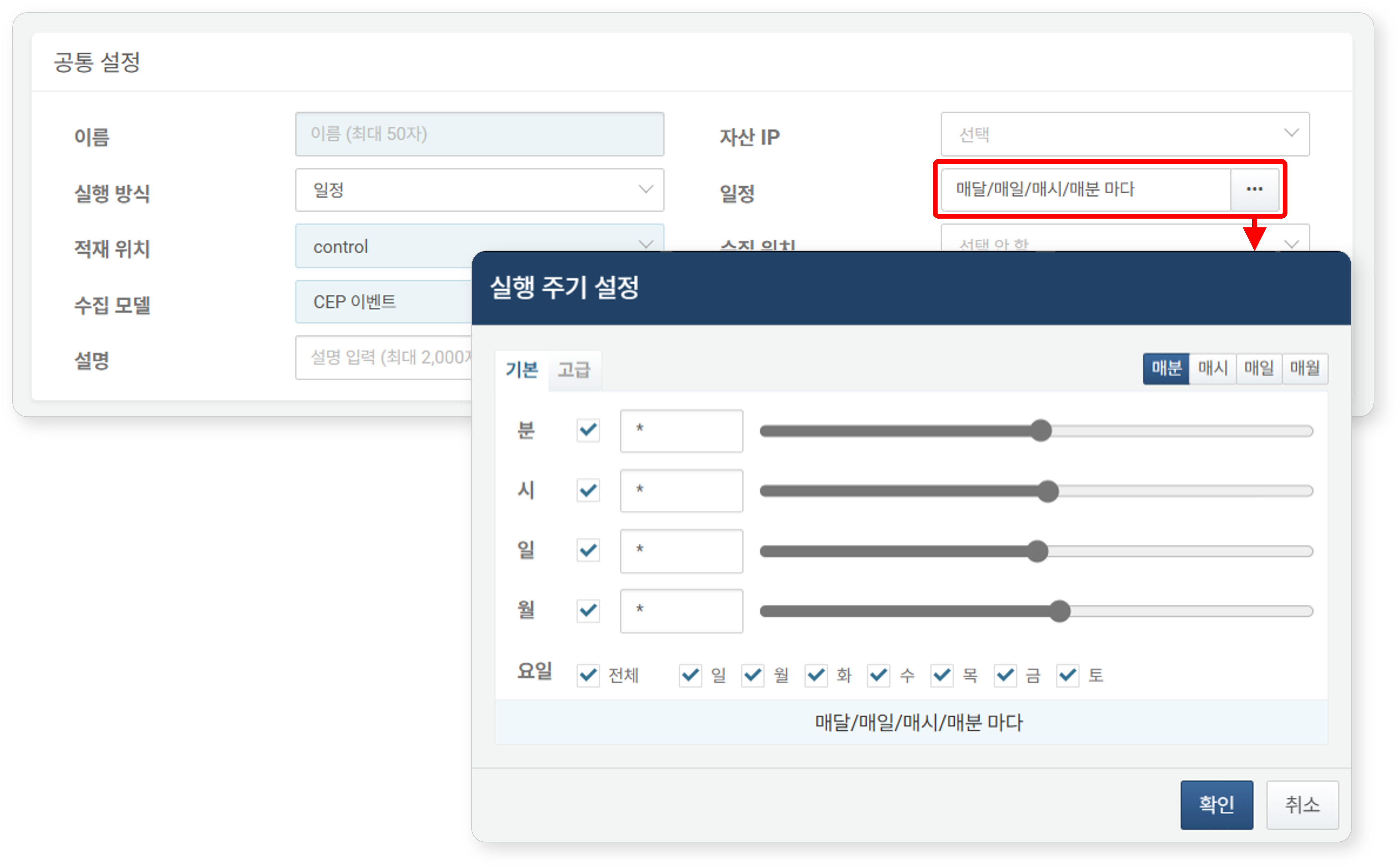
* **Common Settings**: General settings required to configure the logger.
* **Collection Settings**: Settings that must be specified according to the collection method defined in the selected **Logger Model** from the common settings.
* **Filter Settings**: Regular expression patterns to apply when filtering logs to collect or discard from the collected raw data.

The default state of the added logger is disabled. To activate the logger, click the activation button of the added logger.

Common Settings

Common settings consist of general configuration items for the logger, including the data collection period/schedule, the node where the logger will run, the node and table where the collected data will be stored, and the logger model to apply during the normalization process of the collected data. Additionally, it provides functionality to connect with site and asset IP address information necessary for security policy operations.



* **Name**: The name to identify the logger.
* **Asset IP**: Site and asset IP address information for the data source. This item supports searching.
* **Execution Method**: The **Period** or **Schedule** for the active logger to collect data (default: period).
* **Period**: The execution period of the logger specified in seconds (default: 5 seconds).
* 
* **Schedule**: The execution schedule of the logger specified in CRON syntax (default: every month/day/hour/minute).
* 
* The execution method applies to active loggers and does not affect the operation of passive loggers.
* **Load Location**: The node where the collected data will be loaded.
* In the case of a standalone Logpresso Sonar, the name of the node the user is connected to will be displayed.
* In the case of a [Cluster](https://docs.logpresso.comnull) configuration, select the node corresponding to the collection node. If the collection node is redundant, data will be loaded simultaneously to both nodes.
* **Collection Location**: The Sentry or node where the logger will run (default: none selected). This item will only be displayed if the Sentry is connected to the Logpresso Sonar server. This item supports searching.
* **Logger Model**: The logger model to apply to the processing of the collected data.
* Only logger models that can be applied to the selected delivery node or Sentry in the **Collection Location** will appear in the list.
* The collection settings will be determined according to the logger model specified here.
* This item supports searching.
* **Table**: The table where the parsed and normalized data will be stored (default: none). Enter the name of the table added in [Tables](https://docs.logpresso.comnull). If the entered table does not exist, a new table will be added with the specified name.
* **Description**: A description of the logger.

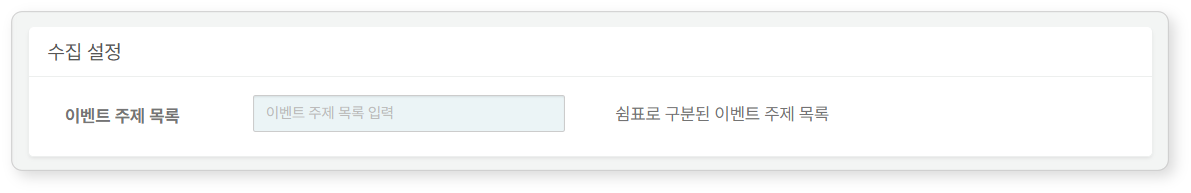
Collection Settings

Collection settings are defined according to the [Type](https://docs.logpresso.comnull) of the **Logger Model** specified in the [Common Settings](https://docs.logpresso.comnull).

The content described here is based on the default loggers of Logpresso Sonar. However, when an app is installed, the necessary logger models for collection are provided, so most cases involve configuring app-based loggers. The collection settings for app loggers are mostly similar to those of default loggers, so please refer to them when configuring app loggers.

CEP Events

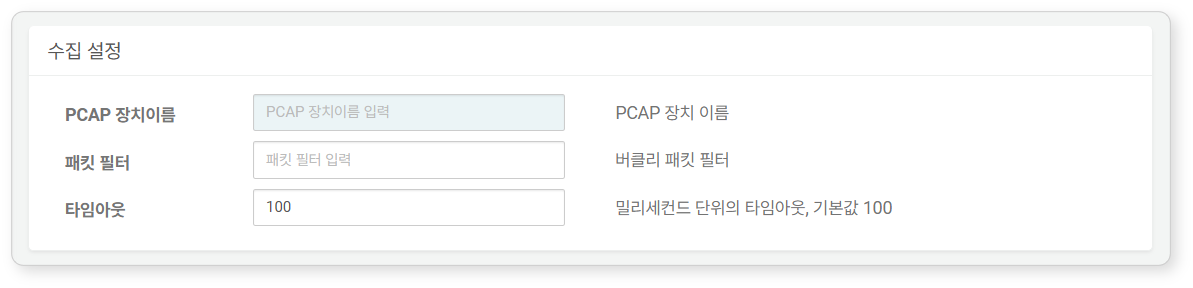
Collects CEP (Complex Event Processing) event data.



* **Event Topic List**: A comma-separated list of event topics. Topics serve a role similar to table names in an in-memory database.

DNS Sniffer

Collects DNS communication packets from mirrored communication packets via a PCAP device.



* **PCAP Device Name**: The PCAP device from which packet data will be collected. Check the name field of the PCAP device list using the [system pcapdevices](https://docs.logpresso.comnull) command and enter it.
* **Packet Filter**: Filter rules to apply for capturing DNS packets. Refer to [PCAP Filter Syntax](https://www.tcpdump.org/manpages/pcap-filter.7.html) for writing.
* **Timeout**: Timeout duration (default: 100 milliseconds).

FTP Directory Watcher

Accesses a network host via FTP communication to collect files that match specified patterns and names in a designated directory.

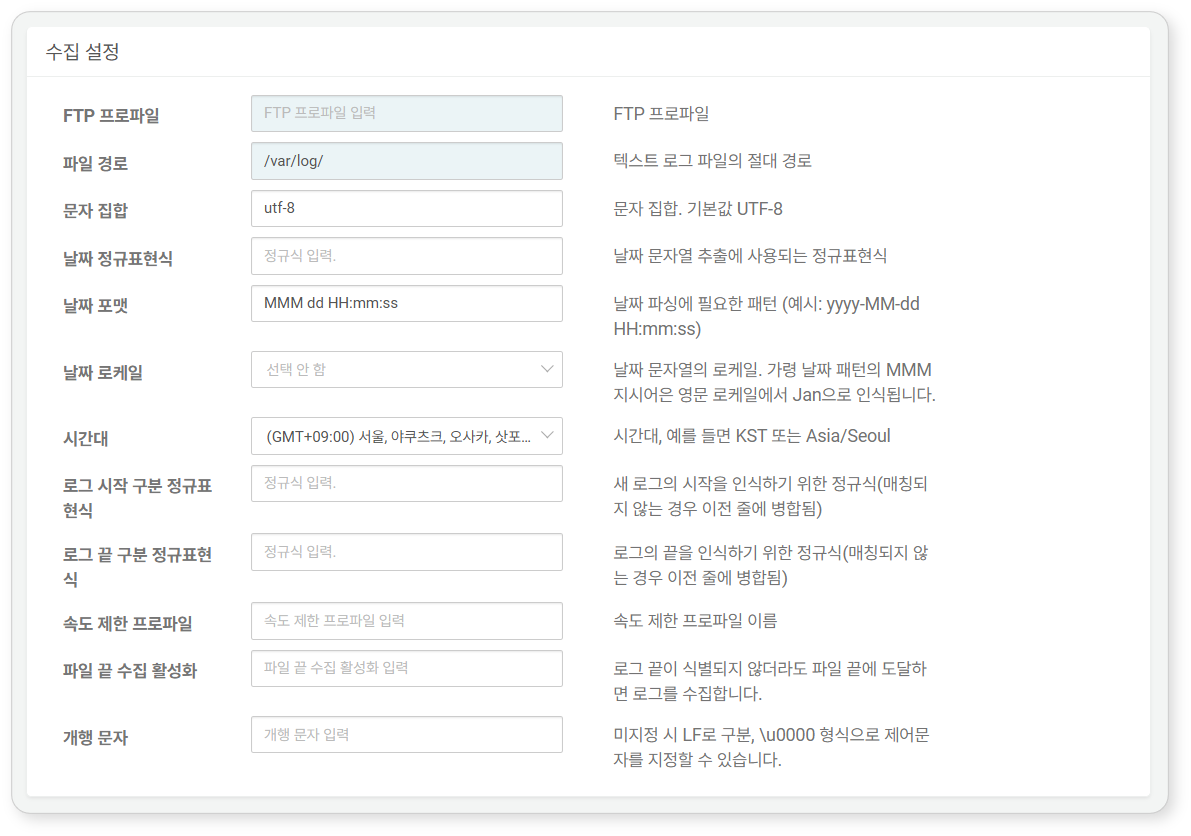


* **FTP Profile**: The [Connect Profile](https://docs.logpresso.comnull) name required for FTP connection to the network host.
* **Directory Path**: The absolute path of the target directory from which files will be collected (e.g., /var/log/foo).
* **File Name Pattern**: A regular expression representing the names of the log files to be collected (e.g., bar\*.gz).
* **Date Regular Expression**: A regular expression to extract date and time information strings from the logs.
* Below is an example of a regular expression for date information used in Apache logs.
* \[(\d{2}/\S{3,10}/\d{4}:\d{2}:\d{2}:\d{2} \+\d{4})\]
* The string groups enclosed in parentheses in the date regular expression are concatenated to create a single date string. This generated date string will be used to extract time information according to the **Date Format** and **Date Locale**.
* **Date Format**: The format to use for parsing the date string (e.g., yyyy-MM-dd HH:mm:ss). The parsing format may vary according to the **Date Locale**.
* **Date Locale**: Select the language to apply to the date string from en, ja, zh, ko (default: none selected).
* **Time Zone**: Select the time zone from the list. When configuring the logger via the API, you can specify the time zone in the format of **KST** or **Asia/Seoul**.
* **Log Start Delimiter Regular Expression**: A regular expression to recognize the start of log records. This is used when a single log record spans multiple lines (if unspecified, newline characters are recognized as log delimiters).
* **Log End Delimiter Regular Expression**: A regular expression to recognize the end of log records. This is used when a single log record spans multiple lines (if unspecified, newline characters are recognized as log delimiters).
* **Character Set**: The character encoding method to apply to the text file (if unspecified, the default is utf-8).
* **Rate Limit Profile**: FTP rate limit profile. The rate limit profile can be created using the logpresso.createFTPRateLimit command in the Logpresso shell.
* **File Name Tag**: The field name to record the collected file name.
* **Path Tag**: The field name to record the directory path.
* **Path Date Extraction Format**: A regular expression to search for date strings in the directory path string. For example, if the directory path contains date information (e.g., /var/log/foo/2022-11-02), the date information will be extracted from the directory path string and used as the collection reference date for the file.
* **Monitoring Target Days**: Only files within the specified number of days after extracting the date from the file path will be collected (default: none, meaning logs will be collected from files without a time limit).
* **Enable End of File Collection**: Collect logs even if the end of the file cannot be identified (if unspecified, the default is false).
* **Newline Character**: Enter the character to use as a newline character in Unicode escape sequence format (e.g., \u000A for LF) (if unspecified, the default is \u000A). For Unicode search, refer to [SYMBL](https://symbl.cc/en/unicode-table/).

FTP Rotated Log Files

Collects rotated log files via FTP communication from a network host.

In Unix or Linux servers, it is common to rename log files periodically for backup and create new log files. Such files are referred to as rotated logs.



* **FTP Profile**: The [Connect Profile](https://docs.logpresso.comnull) name required for FTP connection to the network host.
* **File Path**: The absolute path of the log file to be collected (e.g., /var/log/secure).
* **Character Set**: The character encoding method to apply to the file (if unspecified, the default is utf-8).
* **Date Regular Expression**: A regular expression to extract date and time information strings from the logs.
* Below is an example of a regular expression for date information used in Apache logs.
* \[(\d{2}/\S{3,10}/\d{4}:\d{2}:\d{2}:\d{2} \+\d{4})\]
* The string groups enclosed in parentheses in the date regular expression are concatenated to create a single date string. This generated date string will be used to extract time information according to the **Date Format** and **Date Locale**.
* **Date Format**: The format to use for parsing the date string (e.g., yyyy-MM-dd HH:mm:ss). The parsing format may vary according to the **Date Locale**.
* **Date Locale**: Select the language to apply to the date string from en, ja, zh, ko (default: none selected).
* **Time Zone**: Select the time zone from the list. When configuring the logger via the API, you can specify the time zone in the format of **KST** or **Asia/Seoul**.
* **Log Start Delimiter Regular Expression**: A regular expression to recognize the start of log records. This is used when a single log record spans multiple lines (if unspecified, newline characters are recognized as log delimiters).
* **Log End Delimiter Regular Expression**: A regular expression to recognize the end of log records. This is used when a single log record spans multiple lines (if unspecified, newline characters are recognized as log delimiters).
* **Rate Limit Profile**: FTP rate limit profile. The rate limit profile can be created using the logpresso.createFTPRateLimit command in the Logpresso shell.
* **Enable End of File Collection**: Collect logs even if the end of the file cannot be identified (if unspecified, the default is false).
* **Newline Character**: Enter the character to use as a newline character in Unicode escape sequence format (e.g., \u000A for LF) (if unspecified, the default is \u000A). For Unicode search, refer to [SYMBL](https://symbl.cc/en/unicode-table/).

FTP Daily Directory

Accesses a network host via FTP communication to collect all log files that match the file name pattern while traversing directories created on a daily basis.



* **FTP Profile**: The [Connect Profile](https://docs.logpresso.comnull) name required for FTP connection to the network host.
* **Monitoring Period**: The period for real-time monitoring of file changes (default: 0). When the value is 0, real-time collection is disabled.
* **Directory Path**: The absolute path of the target directory from which files will be collected (e.g., /var/log/foo).
* **File Name Pattern**: A regular expression representing the names of the log files to be collected (e.g., bar\*.gz).
* **Directory Date Regular Expression**: A regular expression to search for date strings in the directory path. If the directory path contains date information (e.g., /var/log/foo/2022-11-02), the date information will be extracted from the directory path string and used.
* **Directory Date Format**: The format to use for parsing the date string (default: yyyyMMdd).
* **Use Directory Date**: Whether to generate log creation dates using the directory date format (default: false).
* **Start Date for Collecting Past Logs**: Enter the start date in yyyyMMdd format. If not specified, collecting past logs will be disabled.
* **End Date for Collecting Past Logs**: Enter the end date in yyyyMMdd format. If not specified, collecting past logs will be disabled.
* **Date Regular Expression**: A regular expression to extract date and time information strings from the logs.
* Below is an example of a regular expression for date information used in Apache logs.
* \[(\d{2}/\S{3,10}/\d{4}:\d{2}:\d{2}:\d{2} \+\d{4})\]
* The string groups enclosed in parentheses in the date regular expression are concatenated to create a single date string. This generated date string will be used to extract time