Page Title: how-to-start-a-live-tail

On this page

How to start a live tail?

Overview

â€‹

The live tail feature enables you to access all your log events in real-time from any source in your infrastructure that is sending the logs to Motadata.

Live tail can be used to check if a new deployment in your environment went smoothly.

Use this feature to see a real-time feed of log events associated with a source. Live tail shows the real time log events in white text visible very clearly over black background making the logs very easy to read.

Navigation

â€‹

Go to Menu, Select

Log Explorer

. After that, Select

Start Live Tail

. The screen to view the live tail is now displayed.

Enter the following details on the Live Tail screen:

Source

Select the source of the log for which you want to view the live tail.

Search Terms

Enter the keyword(s) that you wish to search for from the live tail. Once you have entered a search term, the live tail will only display the log events that have the keyword(s) you specified

Highlight keywords

Enter the keyword(s) you wish to highlight from the live tail. The keyword(s) will be highlighted as shown in the diagram below.

Options On The Live Tail Screen

â€‹

Option

Description

Auto Scroll

Select the check-box to enable auto scrolling of the live tail of log events.

Start streaming/Stop streaming

- Select

 to start the live tail.

- Select

 to stop the live tail.

Clear Logs

Select this button to clear all the logs that have been generated on the screen via live tail.

More Actions

Select the

 button to display the following options:

-

Preferences

 : Select this button to change the text size and line spacing of the log events in the live tail.

-

Split Screen

 : Select this button to view live tail of two log sources at once in the screen.

-

View Full Screen

: Select this button to view the live tail of log events in a full screen.

-

Create Log Parser

: Select this button to start creating a custom log parser.

Page Title: how-to-view-and-analyze-the-logs

On this page

How to view and analyze the logs?

Overview

â€‹

Once you have configured the logging source, the logs are available in the log explorer for you to view and analyze. The Log Explorer provides a comprehensive tool to analyze logs and troubleshoot issues quickly, enabling you to fix any infrastructure problems.

Navigation

â€‹

Go to Menu. Select

Log Explorer

. The Log explorer is now displayed.

Intelligent categorization of logs

â€‹

The Log Explorer categorizes the logs in your infrastructure intelligently to help you locate the logs you need quickly.

The logs are classified based on the

Type

and the

Group

of the logs.

Type

The logs are classified based on the type of source of the logs, i.e., Platform, application, or the device from which the log is generated. For example, some of the categories are Linux, Windows,

and Firewall.

Group

The logs are classified based on the category of logs they belong to. For example, some of the categories are as follows: Linux Syslog, Microsoft IIS Log, and Windows EventLog.

This categorization of logs is available in the log explorer on a panel beside the Main menu. You can click on any category under

Type

or

Group

to reveal the sub-category until you find the log source at the lowest level. You can select this log source to reveal the log details on the

Log Search

screen.

Graphical representation of log count based on the log type

â€‹

The log explorer main screen shows a graphical representation depicting the classification of logs based on their

Type

and the count of log events for each type. This graphical representation is in the form of circles to make the classification of logs and the log counts easy to comprehend with a quick view.

Each circle represents a log type in the graph. There are two important points to understand these graphs.

The larger the diameter of the circle, the higher the count of log event messages for that particular log type.

A circle within a circle represents the hierarchy of the log types. The inner circle is a sub-category of the outer circle.

Important details related to logs

â€‹

A few important details related to log events can be seen on the log explorer screen:

Events per second

: The number of log events per second being sent to the Motadata log explorer from multiple entities in your infrastructure.

Total Events

: The total number of log events being sent to Motadata log explorer from multiple entities in your infrastructure.

Total Alerts

: The total number of active log alerts in your infrastructure.

Advanced Log Investigation

For detailed investigation and search of particular log events, we can use one of the multiple log investigation features available in the Log Explorer.

This can majorly be divided into two categories:

Log Search

Log Analytics

Let us now look into these tools in detail in the next section.

Page Title: log-analytics

On this page

Log Analytics

Overview

â€‹

Log Analytics is a powerful tool within AIOps that allows you to visualize logs from various IT infrastructure components, including applications, servers, and network devices. By providing critical insights into an infrastructure and helping relevant teams quickly identify why an issue occurred, Log Analytics is an essential feature for any organization.

We have seen that the

Metric explorer

tool is used to identify the trends in the metrics in an infrastructure. In a similar fashion, Log Analytics tool is used to identify the trends in the logs in an infrastructure. By allowing to visualize the complex log data, Log Analytics provides critical insights into an infrastructure and helps the relevant teams to quickly and easily identify why an issue occured in their infrastructure.

The parsed logs are available for visualization in Log Analytics, which is an open search query platform that enables you to view log data in multiple visualizations to help you solve a variety of use cases. You can build queries on the Log Analytics screen to visualize data in any of the visualization types mentioned above and use these data visualizations to detect the source of problems in your infrastructure.

You can get insights with the right context on the basis of all the

parsed fields

in the system using log analytics which is an open search query platform. You can view the log data in multiple vizualisations to help you solve a variety of use cases based on the query that you build on the log analytics screen.

Use-Cases

â€‹

You can build queries on the log analytics screen to visualize the data in any of the visualization types mentioned above and use these data visualizations to detect the source of problems in your infrastructure.

Suppose you want to view the logs from a specific source in your infrastructure and the count of the log events that your infrastructure is ingesting from that particular log type you can do so by creating a widget on the log analytics screen.

Suppose you want to view the logs with error messages belonging to a particular severity and you want to view these logs or just learn about their count, you can do so by creating a relevant widget on the Log Analytics screen.

Suppose you want to look at Windows specific logs and we want to filter the log events based on the severity and then we can further group these log events based on the source IP, this enables you to identify the source IP generating the most errors based on the log severity.

*

diagram for query on log analytics along with the graph

*

Navigation

â€‹

Go to Menu, Select

Log Explorer

. The Log explorer is now displayed. Select

Investigate in Search

. The

Log Search

option is displayed by default.

Now, select the

Log Analytics

tab to open the tool.

Select the Visualization

â€‹

First, select the visualization that you want to use on the log analytics tool to display the data.

You can use this tool to plot all the log data ingested via the various default vizualisation options including the following:

Chart

Grid

Top N

Gauge

These are some of the default

vizualisations

that are also available in the Dashboards and Widgets.

After selecting the visualization, we now move to querying data on the widget.

Querying data on the Widget

â€‹

After selecting the visualization for the log data, we now query the data we want to display on the widget.

Select the counter for which you want to display the data on the widget.

Select the aggregate function that you want to be applied on the metric selected.

Select the correct option as per the following:

Source Host

Select this option if you wish to select specific monitor(s) as the source.

Source Type

Select this option if you wish to select one or more type(s) as the source. All the monitors that belong to the selected type(s) will be selected as the source.

Group

Select this option if you wish to select one or more group(s) as the source. All the monitors in the selected group(s) will be selected as the source.

note

In case you do not make any selection, then the data will be queried from all the log sources in the system that have the selected counter.

Select the correct option as per the following:

Option

Description

Select Source

This option is displayed if you select

Source Host

in the previous selection. Select this option to specify the host(s) you want to select as the source. In case you don't specify any host, all the hosts with logs in the system that have the selected counter will be specified as the source.

Select Source Type

This option is displayed if you select

Source Type

in the previous selection. Select this option to specify the source type(s) you want to select as the source. In case you don't specify any type, all the log source types in the system will be specified as the source.

Select Group

This option is displayed if you select

Group

in the previous selection. Select this option to specify the group(s) you want to select as the source. In case you don't specify any group, all the groups with logs in the system will be specified as the source.

Select an option from the

Result By

drop-down if you need to group the data after aggregation.

You can learn more about querying data on the Log Analytics widget in detail

here

.

Saving the visualization as Widget

â€‹

You can save the visualization you created on the Log Analytics tool as a widget from the same screen. You can then use this widget on any of the dashboards you have created or you can use this widget on any future dashboards that you wish to create.

Click on

to save the visualization you created on the log analytics tool as a widget. Enter the

Widget Name

,

Description

, and click on the

Save

button to create the widget right away.

Page Title: log-collector

On this page

Log Collection Profile

Overview

â€‹

With the help of Log collection profile, you can ingest logs remotely. Motadata AIOps is capable of collecting the logs remotely and without an agent installed on the end device. Moreover, the end device also does not necessarily needs to be added as a monitor in order to collect the logs.

Navigation

â€‹

Go to Main Menu, select

Settings

. After that, go to

Log Settings

. Select

Log Collection Profile

.

Log Collection Profile Screen

â€‹

Field

Description

Profile Name

Name of the Log Collection Profile.

Description

Description for the log collection profile.

Collection Type

Protocol used for log collection.

Log Parser

Type of log parser assigned to the collection profile.

Collection Status

Indicates if Motadata AIOps is being able to fetch the logs from then device or not.

Status

Indicates if the logs are being collected or not.

Action

Actions available for a Log Collection Profile.

Now, let's understand how to create a Log Collection Profile.

Create Log Collection Profile

â€‹

On the

Log Collection Profile

screen, click on the

Create Log Collection Profile

option in the top right corner.

Enter the details of all the parameters on the

Create Log Collection Profile

screen as per the following:

Field

Description

Profile Name

Enter the name of the Log Collection profile you want to create.

Description

Provide an appropriate description for the Log collection profile.

Collection Type

Select a protocol for collection of logs using the dropdown menu.

Log Parser

Select a log parser type using the dropdown menu. If you do not select a category, the ingested logs will be showcased under the "Other" category in

Log Explorer

.

Runbook

Select an existing Runbook from the dropdown menu. This runbook will be used for Log Collection.

Create Runbook

Create a Runbook right from this screen if you do not have an existing Runbook for log collection.

Collection interval

Specify the interval between two consecutive log collection.

Timeout

Specify the duration after which Motadata AIOps should stop try to collect logs and abort the operation in the event of failure to reach the device.

Click on

Create Log Collection Profile

to save the changes and create the profile.

To reset fields, click on the

Reset

option.

Page Title: log-forwarder

On this page

Log Forwarder

Overview

â€‹

Log Forwarder in Motadata AIOps allows you to forward the ingested logs to a third-party software. This allows users the flexibility to perform any operations on the collected log data or analyze it using a different software. You can choose between

Syslog-TCP

and

Syslog-UDP

types of log fowarders.

Motadata AIOps can forward logs in either JSON format or raw Logs. You can also filter specific logs by applying a source filter.

Navigation

â€‹

Go to Main Menu, select

Settings

. After that, go to

Log Settings

. Select

Log Forwarder

.

Log Forwarder Screen

â€‹

The following fields are present on the screen:

Field

Description

Forwarder Name

Name of the forwarder.

Description

Description for the forwarder.

Forwarder Type

Type of the forwarder.

Forward As

Format of logs in which they will be forwarded.

Forwarder Status

Status of the Log Forwarder.

Action

Actions available for the Log Forwarder.

Now, let's understand how to create a Log forwarder.

Create Log Forwarder

â€‹

On the Log Forwarder screen, click on the

Create Log Forwarder

option. Since the parameters to create a

Log Forwarder

vary as per the type of Log Forwarder, let's understand how to create each Log Forwarder type individually.

Syslog - TCP

Syslog - UDP

Enter the details of all the parameters on the

Create Log Forwarder

screen as per the following:

Field

Description

Forwarder Name

:

Enter the name of the forwarder you want to create.

Description

:

Provide a description for the log forwarder. You can also describe where the log are being forwarded.

Forwarder Type

:

Select

Syslog-TCP

from the options in dropdown.

Destination IP

:

Enter the destination IP where you want to forward you logs.

Destination Port

:

Enter the destination port number.

Source Filter

:

Select a source filter option using the dropdown menu.

Source

:

Depending on the option you chose in the

Source Filter

field, select an IP, host type, or group from the dropdown menu.

Filter

:

Set a filter condition if you wish to filter out logs before forwarding them. The Filter is discussed in detail further in this guide.

Forward Log as

:

Choose the format in which logs will be forwarded from the dropdown menu.

Once you have configured all the options according to your requirements, click on the

Test

button. Motadata AIOps will then test the Port and IP configuration settings. Once the test is succesfull, the

Create Forwarder

button will appear.

Enter the details of all the parameters on the

Create Log Forwarder

screen as per the following:

Field

Description

Forwarder Name

:

Enter the name of the forwarder you want to create.

Description

:

Provide a description for the log forwarder. You can also describe where the log are being forwarded.

Forwarder Type

:

Select

Syslog-UDP

from the options in dropdown.

Destination IP

:

Enter the destination IP where you want to forward you logs.

Destination Port

:

Enter the destinattion port number.

Source Filter

:

Select a source filter option using the dropdown menu.

Source

:

Depending on the option you chose in the

Source Filter

field, select an IP, host type, or group from the dropdown menu.

Filter

:

Set a filter condition if you wish to filter out logs before forwarding them. The Filter is discussed in detail further in this guide.

Forward Log as

:

Choose the format in which logs will be forwarded from the dropdown menu.

Once you have configured all the options according to your requirements, click on the

Test

button. Motadata AIOps will then perform a ping check. If the test is successful, the

Create Log Forwarder

button will appear.

Configuring Prefilters

â€‹

You can use prefilters to filter out logs before forwarding them. Prefilters allow you to define a specific criteria that must be met before the logs can be forwarded.This ensures you only receive logs that are relevant to your monitoring or investigation needs.

Before we dive into understanding all the configuration parameters available in Prefilter; you should know that a Prefilter allows you to create 3 groups and each group accepts a total of 3 criteria.

Field

Description

Group(s) Matching

:

You can choose the type of operation to be performed on inter-groups. Below is a gist of available options:

-

ALL

:When selected, this will ensure that filtering criteria defined in all groups defined is met.

-

ANY

: When selected, this will ensure that filtering criteria of any ONE group is met.

Group Matching

:

This option will help you define if you want to include or exclude the logs that meet the criteria defined in a single group:

-

Include

: When selected, logs that meet the defined criteria will be forwarded.

-

Exclude

: When selected, logs that meet the defined criteria will be excluded and not forwarded.

Criterias

:

This option will help you define intra-group operations. Below is a gist of available options:

-

All

: If selected, logs will have to meet all the defined criteria in the group.

-

Any

: When selected, logs will have to meet only one of the criteria defined in the group.

Counter

:

Select a metric counter using the dropdown.The options may vary depending on your selection of Forwarder type.

Select Operator

:

Use the dropdown to select operator to be performed on the selected counter.

Value

:

Define a value with respect to the counter and operator to complete the criteria.

To add new group, click on the

Add New Group

option.

To Preview a sample of logs that will be forwarded, click on the

Preview

option. Motadata AIOps will automatically adjust the preview according to the log format you have selected in the

Forward Log As

field. By default, the sample log preview will be of 30 minutes.

Click on

Reset

to reset all the parameters.

Page Title: log-ingestion

On this page

Log Ingestion

Overview

â€‹

In order to explore your logs, you need to configure a logging source to send the logs to Motadata AIOps. The following type of logs can be ingested in Motadata from your infrastructre, some of them directly and some of them through an agent.

Application Logs

Server Logs

Network Logs

Application Logs

Server Logs

Network Logs

Application Logs

â€‹

A variety of applications are used in the current scenario across infrastructure around the world and these applications generate a large volume of log data that can be used for analysis to figure out why a particular issue occured in an application.

Example of Sending Application Logs to AIOps

â€‹

Let us take an example to understand how to send Apache logs to Motadata AIOps.

Install agent on the Apache server.

Go to

Settings

. Select

Monitoring Settings

. After that, select

Agent Monitor Settings

.

Select

View Details

 against the agent for which you want to ingest the logs. Click on the

Log

 tab to start the log configuration.

Enter the log configuration details as follows:

Field

Description

Log Agent Status

Toggle this button

ON/OFF

 to start/stop the log ingestion for this agent.

Log Directory

Enter the exact path where the log file is located on the server. In this case for apache logs on the linux server we can see that the path is entered as '

/var/log/apache2/

'. This is the path where the Apache logs are located on the server.

Log Include

Mention the file name or the extension of the file in this field to make sure that only the logs from that particular file present at the path mentioned in

Log Directory

 above are ingested in the system. For example, if you want to ingest logs from a file

'access.log'

you can do that by mentioning the file name in this field as

'access.log'

. You can ingest logs from all the log files with the extension

'.log'

by specifiying

'*.log'

in this field.

Multiline Log

Use this toggle button to specify that the log you are ingesting are multiline logs.

File Pattern

This field is available only if you switch the

Multiline Log

toggle button ON. Specify the file from which you want to ingest the log data.

Log Pattern

This field is available only if you switch the

Multiline Log

toggle button ON.Specify the log pattern of the multiline logs that you want to ingest from the file that you have specified in the previous field. Enter the regex that could be used to identify the pattern that is used to differentiate two lines of logs in multiline logs.

We just looked into how you can ingest Apache logs into Motadata AIOps by just pointing AIOps towards the log file by providing the directory and log file details in the log configuration in the agent. Similarly, you can also do the same for other application logs such as IIS, NGINX, and many more.

Example of Ingesting Multiline Logs in Motadata AIOps

â€‹

Ingesting multiline logs in Motadata AIOps requires identifying the start of each new log entry within the multiline logs. This is achieved using a specific log pattern that can be defined using regular

expressions (regex). Let's take an example of multiline IBM MQ logs to illustrate this process.

IBM MQ Multiline Logs Example

â€‹

Consider IBM MQ logs where each new log entry starts with a line similar to:

04/09/24 09:46:51 - Process(9241016.5111) User(mqm) Program(amqrmppa)

In these logs, each new entry begins with a timestamp followed by the word "Process". This pattern can be utilized to indicate the beginning of a new log entry.

Defining the Log Pattern

â€‹

To help AIOps correctly ingest and parse these multiline logs, we need to define a regex pattern that matches the start of each new log entry. For our example, the regex would look like this:

\d+\/\d+\/\d+\s+\d+:\d+:\d+\s+-\s+Process

This regex pattern matches:

A date in the format

MM/DD/YY

(e.g.,

04/09/24

)

A time in the format

HH:MM:SS

(e.g.,

09:46:51

)

The literal string

- Process

Configuring the Log Pattern in Motadata AIOps

â€‹

In the

Log Pattern

field, enter the regex pattern:

\d+\/\d+\/\d+\s+\d+:\d+:\d+\s+-\s+Process

.

By configuring the log pattern with this regex, Motadata AIOps will be able to recognize the start of each new log entry in the multiline IBM MQ logs. This allows AIOps to accurately ingest and parse the logs, ensuring that the log entries are correctly identified and processed.

This approach can be applied to other types of multiline logs by defining an appropriate regex pattern that matches the unique start of each log entry for those logs.

Server Logs

â€‹

There are multiple servers in infrastructure setups used by IT teams to provide users access to a variety of services and applications. These servers in turn are accessed by a range of users who use it for a variety of purposes

For example, a web server might contain a log of page requests that users might have made. Apart from that there might be multiple requests including access logs and error logs. You might even want to analyse the Syslog from linux servers.

All of these server logs can be ingested by AIOps. We will now take examples of Linux Syslog and Windows Event Logs to see how we can send the server logs to AIOps.

Example of Sending Linux Server logs to AIOps

â€‹

Let us take an example to understand how to send Syslog from Linux Server to Motadata AIOps.

Log in to the Linux Server for which you want to send the Syslog to AIOps.

Open the rsyslog.conf file which is typically located in

'/etc'

.

Look for the following text in the file:

$IncludeConfig /etc/rsyslog.d/*.conf

This text is typically located at the end of the file.

Now we need to provide the AIOps server details to send the Syslog to the AIOps server. In order to do that we need to enter the Motadata server IP below the text located in the step above so now the text looks as follows:

$IncludeConfig /etc/rsyslog.d/*.conf

. @ServerIP:PortNumber

where '

ServerIP

' is the IP address of the Motadata AIOps server and '

PortNumber

' is the log forwarding port number on the linux server.

note

Write @ServerIP to send UDP logs.

note

Write @@ServerIP to send TCP logs.

Now, restart the rsyslog service to start sending the Syslog to Motadata AIOps.

Example of Sending Windows Server logs to AIOps

â€‹

Now, let us see how we can send the Windows event logs to AIOps.

Install agent

on the Windows server.

Go to

Settings

. Select

Monitoring Settings

. After that, select

Agent Monitor Settings

.

Select

View Details

against the agent for which you want to ingest the logs. Click on the

Log

tab to start the log configuration.

Enter the log configuration details for Windows event logs as follows:

Field

Description

Name

Specify the type of Windows event log that you want to send to AIOps i.e., Application, Security, or System.

Levels

Specify the event level of the Windows event log from the dropdown that you want to send to AIOps i.e., Trace, Critical, Error, Warning, Informational, or Verbose.

Events

Specify the Event Id of the Windows event log that you want to send to AIOps.

note

In case you do not specify any specific event level, then logs of ALL the event levels from the type of log selected will be sent to AIOps. Similarly, if you do not specify any Event ID, then ALL the logs from the selected log type and the selected event level will be sent to AIOps. For Example, if you specify the

Name

as

Application

and

Levels

as

Critical

, then ALL the critical application logs will be sent to AIOps.

We just looked into how you can ingest Windows event logs into Motadata AIOps. Now, let us look into how you can send network logs to AIOps.

Network Logs

â€‹

Just like other types of logs, network logs are a valuable source of information for maintaining the performance and security of your IT infrastructure. In Motadata AIOps, ingesting network logs follows a familiar process as sending syslog data.

By utilizing the same method employed for sending syslog, you can seamlessly incorporate network logs into Motadata AIOps for comprehensive monitoring and analysis. This approach ensures consistency and ease of use when dealing with different types of log sources.

Let's explore how to ingest network logs into Motadata AIOps using the established method.

Page Title: log-inventory

On this page

Log Inventory

Overview

â€‹

Motadata AIOps provides you the facility to specify the type of the log that arrives from an unknown source in advance. This enables the system to identify the logs coming from the unknown source and assign appropriate parser to the log source so that the incoming logs can be parsed correctly and ultimately placed into the appropriate category.

When the logs come from an unknown source, any one of the following two options can happen:

In case you have already configured the source in the log inventory and assigned a parser to the source, the logs will be parsed with the selected parser and the logs will move into the right category based on the

Type

of the device configured and the parser assigned to the device in the log inventory .

In case the source is not configured in the log inventory, the logs arriving from that source will be placed in the

'Other'

category. A record corresponding to this unknown log source is created in the

Log Inventory

when the logs arrive from this unknown source. Once we assign type to this unknown log source and assign a parser to this log source in the log inventory, the new logs that arrive from the same source from that moment onwards will be parsed with the selected parser and the logs will move into the right category based on the

Type

of the device configured in the log inventory.

note

You can create a record corresponding to the unknown log source in advance in the

Log Inventory

to ensure that the logs that arrive from the source are parsed right away and are not moved into the

'Other'

category.

Navigation

â€‹

Go to the Main Menu, Select

Settings

. After that, go to

Log

. Select

Log Inventory

. The log inventory list is now displayed.

Log Inventory Screen

â€‹

The following fields are available on the Log Inventory screen:

Field

Description

Source

The IP address of the source.

Source Type

The type of the log source.

Assigned Parsers

The count of parsers assigned to the log source.

Category

Select the category that best describes the type of logs you are ingesting from the log source. This helps in organizing and managing different types of logs effectively.

Group

Select the appropriate group for the log source.

Actions

We will discuss all the actions available for log inventory in detail below.

Assign Log Parser

â€‹

You can assign a log parser to a log source in the log inventory. When a log parser is assigned to a log source, all the logs from that source are then parsed using the parser assigned.

You should assign a parser to an unknown log source by creating a record in the log inventory for that source in advance if you know that you will be receiving logs from that source in the future. In case you do not create a record in the log inventory for this unknown source, a record will be created for you in the log inventory when the logs are received in the AIOps server. These logs will be placed in the

'Other'

category.

Once we assign

Type

to this unknown log source and assign a parser to this log source in the log inventory, the new logs that arrive from the same source from that moment onwards will be parsed with the selected parser and the logs will move into the right category based on the

Type

of the device configured in the log inventory.

Click on

against the log source to which you want to assign a log parser. Select

Assign Log Parser

to display a list of parsers you can assign to the log source.

Now, check the box against the parser(s) that you want to assign to the log source and then click on

Assign Log Parser

button to assign the parser to the log source.

Remove Assigned Log Parser

â€‹

You can also remove the parsers already assigned to a log source. The logs that are already parsed will remain parsed with the same parser even if you remove the assigned parser.

Click on

against the log source for which you want to remove a log parser. Select

Remove Assigned Log Parser

to display the list of parsers assigned to the log source.

Now, check the box against the parser(s) that you want to remove from the log source and then click on

Unassign Log Parser

button to remove the assigned parser from the log source.

Edit Log Inventory

â€‹

You can edit the details of a log source in the log inventory.

Click on

against the log source that you want to edit. Select

Edit Log Inventory

. A pop-up is displayed on the right side of the screen with all the details of the log source.

Now you can edit the log source details on this screen.

Assign Log Source Time Zone

â€‹

When ingesting logs into Motadata AIOps, itâ€™s essential to consider the time zone of the log source. If the logs originate from a different time zone than the one where the user is currently logged into the system, specifying the correct time zone for the log source is crucial to avoid any confusion and ensure accurate log analysis.

Steps to Assign Log Source Time Zone

â€‹

To assign the time zone of the log source in the log inventory, follow these steps:

Navigate to the Log Source in the Log Inventory for which you need to assign the timezone.

Click on

against the log source that you want to edit. Select

Edit Log Inventory

.

In the edit inventory screen, locate the field labeled

Source Time Zone

. Select the appropriate time zone for the log source from the dropdown menu.

Select

Update Log Inventory

to save the changes and update the log inventory.

When to Assign a Log Source Time Zone?

â€‹

Different Time Zones:

Assign the log source time zone if the logs are being ingested from a different time zone than the userâ€™s local time zone.

Same Time Zones:

If the log source and the user are in the same time zone, NO action is required as there will be no time zone conflict.

By accurately assigning the log source time zone, you ensure that the timestamps in the logs are correctly interpreted and displayed, avoiding any potential confusion that may arise from time zone differences. This practice helps in maintaining precise log data analysis and improves the overall effectiveness of log monitoring within Motadata AIOps.

Create Record for an Unkown Log Source in the Log Inventory

â€‹

You should assign a parser to an unknown log source by creating a record in the log inventory corresponding to that source in advance if you know that you will be receiving logs from that source in the future. The assisgned parser will then be used to parse the logs received from this unkown log source.

Click on

to create a record for an unkown log source in the inventory.

Enter the following details on the Log Inventory screen:

Field

Description

Source

Enter the IP address of the unknown source from which you expect to receive logs in the future.

Type

Select the type of the source device from the drop-down.

Groups

Select the group to which the source device belongs.

Log Parser Name

Select the log parser that you want to assign to the log source.

After entering all the details, click on

Create Log Inventory

to create the record corresponding to the log source in the inventory and assign a log parser to the source.

Select

Reset

to erase all the current field values entered in the pop-up, if required.

Page Title: log-mechanism

On this page

Log Mechanism

Overview

â€‹

In order to make the best use of the AIOps log analysis capabilities, let us start by understanding the log mechanism in AIOps.

The log mechanism in AIOps includes the following major steps:

Log Ingestion

: The raw logs are first ingested in AIOps. There are multiple ways to

configure a logging source.

. The first step in log monitoring is log ingestion, which involves collecting the raw logs from various sources and feeding them into the AIOps system. There are several ways to configure a logging source, including syslog, and agents. Log ingestion can be done in real-time or through batch processing. Real-time ingestion involves processing the logs as they are generated, while batch processing involves processing the logs at specified intervals.

Log Parsing and Filtering

: Once the logs are ingested into the AIOps system, the logs are

parsed

and filtered based on the parser assigned to the source device. The parser identifies the log format and extracts the relevant information from the log messages. The extracted information is then stored in a structured format for further analysis.

There are several inbuilt log parsers available in AIOps, but users can also create their own parsers to parse logs from specific devices or applications that may not be supported by the inbuilt parsers.

Log Dump in Database

: After the logs are parsed, the parsed logs are stored in a database. The database is usually structured in a way that is optimized for log storage and retrieval. The logs are categorized based on the parser assigned to the log source, making it easy to locate and analyze specific logs.

Log Exploration and Analysis

: The final step in the log mechanism is

Log exploration and analysis

. Once the logs are in the database, they can be explored and analyzed using Log Explorer. Log exploration and analysis involves searching and filtering logs, creating dashboards and reports, and identifying patterns and trends in the log data. Motadata AIOps uses machine learning and AI algorithms to analyze log data and identify anomalies or potential issues. This can help organizations proactively address issues before they become major problems.

We will now look into all the steps in detail in the next sections.

Page Title: log-parsing

On this page

Log Parsing

Overview

â€‹

Once the logs are ingested into AIOps, the next step is filtering and parsing these logs to make them meaningful to the users. Log parsing is the process of breaking down the log data into its constituent parts and assigning a meaning to it. Once the logs are parsed, users can get meaningful insights from these logs when viewed using the AIOps Log Explorer.

AIOps uses log parsers to parse the incoming log data. A log parser is a software component that can extract data fields from a log message, normalize and format them, and categorize them into various fields.

Let us understand how the Log parsing and Filtering take place in detail.

Once the logs are sent to AIOps, they could either be from a known source (Monitor) or an unknown source (a device that is not a Monitor in AIOps).

When the logs come from a known source (Monitor), a parser is assigned to the source based on the

Type

of the source. After that, the logs are parsed in the system and moved into the appropriate log category. After that, the logs are available in the system in a suitable format for analysis.

When the logs come from an unknown source, any one of the following two options can happen:

In case you have already configured the source in the log inventory and assigned a parser to the source, the logs will be parsed with the selected parser and the logs will move into the right category based on the

Type

of the device configured and the parser assigned to the device in the log inventory .

In case the source is not configured in the log inventory, the logs arriving from that source will be placed in the

'Other'

category. A record corresponding to this unknown log source is created in the

Log Inventory

when the logs arrive from this unknown source. Once we assign type to this unknown log source and assign a parser to this log source in the log inventory, the new logs that arrive from the same source from that moment onwards will be parsed with the selected parser and the logs will move into the right category based on the

Type

of the device configured in the log inventory.

note

You can create a record corresponding to the unknown log source in advance in the

Log Inventory

to ensure that the logs that arrive from the source are parsed right away and are not moved into the

'Other'

category.

Log Parser Library

â€‹

Motadata AIOps provides an inbuilt library of parsers that are used to parse logs from a range of sources, across multiple vendors. There are 20 inbuilt parsers available in the system out of which 15 are java based parsers and the remaining 5 are regex based parsers.

These parsers are used to turn the raw logs into meaningful data by educating the system about the type of logs, the important fields present in the logs, where exactly these fields are placed in the logs.

Motadata AIOps has inbuilt parsers for several commonly used applications such as Apache, IIS,

NGINX, MySQL, Oracle, and many others. These parsers can also parse logs from various network devices such as firewalls, routers, switches, load balancers, and more.

In case one of the multiple inbuilt parsers is not able to parse some specific logs from a new vendor in the market not supported by AIOps, you can also create a parser of your own to parse these logs. This can be done easily by writing custom rules for the parser. Once the custom parser is created, it can be added to the system and used for parsing the logs.

By using the Log Parser Library in Motadata AIOps, users can parse and filter logs easily, enabling them to identify and troubleshoot issues in their environment with ease.

Navigation

â€‹

Go to the Main Menu, Select

Settings

. After that, go to

Log

. Select

Log Parser Library

. The list of all the log parsers in the system is now displayed.

Log Parser Library Screen

â€‹

The following fields are available on the log parser library screen:

Field

Description

Log Parser Name

The name of the parser.

Used Counts

The number of devices to which the parser is assigned.

Log

The sample log message used to create the parser.

Log Parser Type

The type of log parser.

Actions

The actions available for a parser. We will discuss this in detail below.

Now, let us look into how to create a parser of your own.

Create Log Parser

â€‹

In case one of the multiple inbuilt parsers is not able to parse some specific logs from a new vendor in the market not supported by AIOps, you can also create a parser of your own to parse these logs.

Navigation

â€‹

Go to the Main Menu, Select

Settings

. After that, go to

Log

. Select

Log Parser Library

. The list of all the log parsers in the system is now displayed.

Select

to create a log parser. The screen to create a new log parser is now displayed.

Create Log Parser Parameters

â€‹

Enter the details of all the parameters on the

Create Log Parser

screen as per the following details:

Field

Description

Log Parser Name

Enter the name of the parser you want to create.

Log Parser Type

Select the

parser type

from the dropdown as per the type of parser you wish to create.

Type

Select the type of logs you want to parse. In case you want to add a

type

of log that is not already available in the list, you can do so by clicking on

option.

Vendor

Select the vendor of the device generating the logs you want to parse.

Log Parsing Condition

This field works with the next field

Log Parsing Filters

for filtering logs for parsing based on keywords. The logs filtered through these fields will be parsed

using the parser and the logs that do not get filtered will be moved to the

'Other'

category. Select one of the following two options

-

All

: The log event will be filtered for parsing only if ALL the keywords specified in the next field

Log Parsing Filters

are present in the log event

-

Any

: The log event will be filtered for parsing if ANY of the keywords specified in the next field

Log Parsing Filters

are present in the log event.

Log Parsing Filters, If Log Contains

Specify the keywords to filter the log events. This field works with the previous field

Log Parsing Condition

to filter the log events based on the keywords mentioned in this field.

File Upload

You can use this field to upload the log file that you wish to parse. You can use the logs from this file to create the parser. This process is explained further below.

Log

Enter a sample log event from the log file that you wish to parse.

Regex

This is the auto-generated regex which is created when you click on the fields in the sample log event entered in the above

Log

field. The process to create an auto-generated regex is explained further below.

Fields

This is list of all the fields parsed from the sample log file. The list of the fields is dynamic and is updated based on the changes you make to the parser.

Add Operation

Click on this button to add a custom field to the parser using a concatenation operation between any

two fields that you have seperated using the parser.

Select Plugin

This field ony shows up if you select

Custom Plugin

in the

Log Parser Type

field. Select one of the inbuilt plugins or a custom plugin that you want to use for parsing while using the

Custom Plugin

parser type.

Delimiter

This field ony shows up if you select

Delimiter

in the

Log Parser Type

field. Specify the

delimiter

that you want to use to seperate the fields in the logs

Parser Creation Example

â€‹

Let us take an example. Suppose you want to create a parser for Linux Syslog that parses the timestamp, process, and the process ID from all the messages in the Linux Syslog. We can use a

regex parser

Enter all the details such as the parser name and start creating the parser.

Filter Conditions

â€‹

Suppose you want the parser you are creating to filter and only parse the log events received in the

month of February OR March, you can use the filter conditions in the parser to do the same.

Enter the keywords

Feb

and

Mar

in the

Log Parsing Filters

field to filter the log events that have the timestamp for either February OR March.

Select

Any

in the

Log Parsing Condition

to specify log events containing either

Feb

OR

Mar

will be filtered.

All other log events that do not contain the either of the keywords

Feb

OR

Mar

will be moved to the

'Other'

category.

Suppose you want the parser you are creating to filter and only parse the logs received in March for a particular process ID, 21992 , you can use the filter conditions in the parser to do the same.

Enter the keywords

Mar

and

21992

in the

Log Parsing Filters

field to filter the log events that have the timestamp for March and have the .

Select

ALL

in the

Log Parsing Condition

to specify that the log events containing BOTH the keywords

Mar

and

21992

will be filtered .

All the other log events that do not have BOTH the keywords

Mar

and

21992

will be moved to the

'Other'

category.

Generating the Regex

â€‹

Enter a sample log event from the log file you want to parse in the

Log

field.

Select the part/field of the log event that you want to identify and seperate using the parser. A regex corresponding to the selected part of the log event will be auto generated in the field

Regex

.

For example, you can select the specific part of the sample log event that contains process name to auto-generate corresponding regex to identify process names.

note

The time stamp from the log event is parsed by default and does not require to be added to the regex.

As you select more fields/parts from the sample log event, the regex get modified to include the fields you have selected. For example, after generating the regex for process name, you might also want to include the part of the log event that includes the process ID.

Now, the regex to identify and parse the

Process ID, Process Name, and the Timestamp

is generated.

Provide Field Names and Create the Parser

â€‹

After parsing the field names using the regex, we provide the names to the parsed fields. The names that you provide to the field will be used to identify them in the

Log Search

to view the log details.

For example, in the diagram below, the fields have been named

Process ID, Process Name, and the timestamp

.

Select

Create Parser

to create the parser and add it to the

Log Parser Library

.

Select

Create Parser & Upload Logs

 to create the parser and add it to the

Log Parser Library

 and upload the attached logs to the system. The logs uploaded will be parsed and available to view in the

Log Search

Select

Reset

 to erase all the current field values, if required.

All the logs parsed with the parser you created will be available in the

Log Search

 in the

Linux

 category with the parsed values of

Process ID, Process Name, and Timestamp

.

Page Title: log-search

On this page

Log Search

Overview

â€‹

The log search feature enables you to narrow down to the exact log event that you wish to view and analyze. You can set a condition to filter out the log data to the values you desire to view. Select

 to set the filter condition as per your requirement view the log events you have searched for.

For example, you can set a filter condition to view all the logs from a particular source host for a specific time period by providing appropriate filter conditions. You can then view and analyze the log events you have searched.

Navigation

â€‹

Go to Menu, Select Log Explorer

. After that, select

 .The screen to search and view the log details is now displayed.

Log Investigation using Log Search

â€‹

Count of Log events

â€‹

The bar graph at the top of the

Log Search

 screen displays the count of log events received at different times during the day. To view more details about a specific time period, hover your cursor over the corresponding section of the graph.

To understand other elements of the log search, navigate to the list of tabs present below the bar graph. Select

Event Log

to start with.

Event Log

â€‹

Event Log enables you to view the details of the live log events in your infrastructure.

The timestamp of the log events and the message associated with the event are displayed on this tab by default. You can add more fields from the list of available fields to view the details of that field for each log event. Suppose you want to view the log message along with the host generating these messages, you can simply add the

source.host

field from the list of

Available Fields

to the list of

Selected Fields.

Go to the list of

Available Fields

. Hover the mouse cursor over the field that you want to add in the

Event Log

. Select

present beside the field. This would move the field from the list of

Available Fields

to the list of

Selected Fields

.

Select the

Raw Log

checkbox to show the unparsed version of logs in the

Message

column as received in Motadata AIOps.

Surrounding Logs

â€‹

Under the

Event Log

tab, navigate to a specific log message and select the

View Surrounding Events

button to view the log messages surrounding the selected log message.

Surrounding logs provide a more comprehensive understanding of the environment and help in performing effective root cause analysis, troubleshooting, and incident response.

When an event or alert is triggered, you can view the surrounding logs that capture additional log entries from relevant systems, applications, or infrastructure components. The idea is to gather a broader set of information that might be associated with the event, allowing IT operations teams to have a more complete picture of what occurred.

Organized Log

â€‹

This tab shows all the parsed data from logs in an organized manner. For each log event, the details of all the available fields are shown in a tabular manner. This view enables you to easily skim through the log event data and gather relevant information at a glance.

Creating Reports from Log Search

â€‹

The Log Search feature in Motadata AIOps also allows users to create detailed reports based on specified log filter criteria.

Once you have specified the filter criteria to narrow down the log events you wish to include in the report, click on

Save as Report

.

A dialog box will appear prompting you to enter the report

Name

and

Description

. Provide the required information and click

Save

.

The

Report

will be available to view and analyse further in the

All Reports

category under the

Log

tab. For easy accessibility, you can also mark reports as favourite by clicking on the star icon preceding each of them. All marked reports will be listed under the

My Favourite Reports

section.

Log Event Timeline

â€‹

By default, the Log Search screen displays log events generated on the current day.

This means the log events generated on the current day are shown by default. For example, if the current date is 1st January and the time period is selected as

Today

, then the log events generated on 1st January are shown on the screen.

You can also view the historical log events by changing the time period as required. Click on the

button at the top-right corner of the screen to do so.

Navigation

â€‹

Go to Menu, Select Log Explorer

. After that, select

.The screen to search and view the log details is now displayed. Select the

Pattern

tab to display the tab for pattern correlation.

Use Case 1: Identifying Noisy Patterns - Brute Force Attacks

â€‹

A web-server/application is experiencing a surge in login attempts, indicating a potential brute force attack. The logs from various sources are being ingested into Motadata AIOps, and identifying unauthorized access swiftly is crucial.

Solution with Pattern Correlation

Pattern Identification

Motadata AIOps leverages Log Pattern Matching to identify patterns related to repeated failed login attempts, showcasing a potential brute force attack. Logs with similar patterns, but potentially masked usernames or IPs, are grouped together.

Detection of Irregularities

You will be able to identify the irregularity as an anomaly, as it deviates from regular login behavior.

Efficient Troubleshooting

IT operators can focus specifically on logs related to the identified pattern, making troubleshooting more efficient. By analyzing the masked usernames or IPs, security teams can narrow down the scope of the attack and take appropriate measures.

Reduced Noise

Log Pattern Matching filters out unrelated logs, ensuring that only logs related to the brute force attack pattern are presented. This reduces noise and allows security teams to concentrate on addressing the security threat.

Enhanced Visibility

Security teams gain enhanced visibility into the attack pattern, enabling them to understand the scale and tactics of the brute force attack.This insight accelerates the response time and helps in implementing necessary security measures.

Use Case 2: Outlier Detection in Critical Events

â€‹

An organization's IT infrastructure is generating a massive volume of logs, including rare errors or critical events. Detecting outliers within this ocean of logs is essential to identify potential issues that may have a significant impact.

Solution with Pattern Correlation

Pattern Identification

Motadata AIOps employs Log Pattern Matching to identify patterns associated with rare errors or critical events. Logs with similar patterns are grouped together, indicating potential outliers in the log data.

Detection of Irregularities

You will be able to recognise patterns associated with rare events as outliers, as they occur infrequently compared to regular log patterns.

Efficient Troubleshooting

IT operators can focus specifically on logs related to the identified outlier patterns, streamlining the troubleshooting process. By examining the logs identified as a outlier, teams can quickly identify and address critical issues that may impact the overall system.

Reduced Noise

Log Pattern Matching filters out logs that do not match the identified outlier patterns, reducing noise and highlighting only the logs of interest. This ensures that IT teams prioritize their efforts on critical events rather than going through a vast amount of log data.

Enhanced Visibility

IT administrators gain enhanced visibility into rare errors or critical events by analyzing logs with

identified outlier patterns. This visibility enables proactive measures to be taken to prevent potential issues from escalating and impacting the organization's IT environment.

In the Log Pattern Matching tab, users can explore different patterns, each displaying the count of logs belonging to that pattern and the percentage of those logs out of the total logs ingested. This interface provides a comprehensive view of how logs are correlated and grouped based on their patterns, offering users valuable insights into their log data.

Page Title: overview

On this page

Log Management

Overview

â€‹

Log management is an essential aspect of modern IT infrastructure that enables you to understand the "why" of events that occur on your network. While metrics can help answer "what," logs provide detailed insights into the context and reasons behind the metrics.

However, modern infrastructure generates a massive volume of log data, making it challenging to store, analyze, and make sense of. It is also essential to have all log data available in a centralized location for efficient log management.

Motadata Log Explorer offers an out-of-the-box solution to dynamically parse and visualize millions of lines of log data. You can view the live tail of logs being generated in real-time and quickly identify specific logs through machine learning-powered pattern matching. The tool also allows you to create widgets for frequently viewed log data, streamlining your log management process.

With the Log Explorer, you can easily troubleshoot issues and reduce context switching by identifying logs through intelligent categorization. You can also view the surrounding logs related to a specific log event with just one click, making it easier to understand the root cause of issues.

Dynamically parse and visualize millions of lines of log data with out-of-the-box inbuilt parsers.

View the live tail of the logs being generated in real-time.

View the surrounding logs related to a certain log event with just one click.

Identify specific logs through machine learning powered pattern matching for logs.

Search and highlight required keywords from the live tail.

It makes life easier by collecting, processing and centralizing logs into a log explorer.

Allows quick troubleshooting and reduces context switching by easily identifying logs through intelligent categorization of logs.

Allows to create widgets on the go for specific log data that you wish to view repeatedly.

Overall, the Motadata Log Explorer makes log management more accessible and efficient by collecting, processing, and centralizing logs into a single platform.

Page Title: types-of-parsers

On this page

Types of Log Parsers

Overview

â€‹

Parsers are crucial for processing of logs. They facilitate extraction of selected information which can be further used for diagnostics and optimizations of a network or a system. Motadata AIOps allows you to parse logs using four distinct methods which are explained as follows:

Regex

â€‹

Creating a Regex for log is very simple. You can either upload a log file using the

Upload

option or enter a sample log in the

Log

field. Then, select a part of the log and Motadata AIOps will automatically create a regex of the selected portion for you. You will only need to enter the

Field name

and provide a

Field Type

for the values. You can also edit the created Regex using the edit option next to the created regex.

note

The Timestamp is usually selected automatically when you select the sample log. In a rare case Motadata AIOps does not recognize it automatically, just double-click on the Timestamp field to create a regex for the field.

JSON

â€‹

JSON type log files consists information in

Key:Value

pairs. The JSON log parser in Motadata AIOps extracts the information and segrates the

Key

and

Value

. Every

Key:Value

pair from the log is extracted and segregated by the parser into

JSON Key

and

JSON Value

fields.

Once you upload the log file using the Upload option or enter a sample log in Log field. AIOps will parse the logs and display the segregated key and value pairs. You can then select these

Key:Value

pairs and add them as

Field Name

and

Field Value

respectively by selecting them using individual checkbox. You would then need to define the

Field Type

for each of the added values.

Do remember, should you want to parse a different combination of fields or

key:value

pairs, you can always create a separate parser. For instance, you need complete log information (error ID, error message, timestamp, and other applicable fields.

) but also want to parse two fields (

error ID and error message

) separately from the same set of logs, create two parsers. One that parses the complete log and the other one that only parses the specific fields.

You can also concat (join) two defined fields and create a custom field.To create a custom fied, click on

Add Operation

, then select a

Field Value

 from a dropdown, choose the CONCAT operator using the

Operator

 dropdown and finally choose the second

Field Value

. Click on

Add Field

 to add the custom field. Do remember you will need to define the

Field name

 and

Field value

 for the concated field.

Delimiter

â€‹

Delimiter parser uses a special character (

comma, colon, or semicolon

) to separate different portions of a log and extract information. You need to upload a log file using the

Upload

option or enter a sample log in the

Log

field. Also, when parsing the logs using Delimiter, you will need to mention the special character in the

Delimiter

field.

note

Ensure that you enter only a single character in the

Delimiter

field. Entering multiple characters at the same time may produce unexpected results.

Once the sample log is processed and fields are generated with the help of your mentioned delimiter; select each one that you wish to parse by clicking on the individual checkboxes preceding each field of the log. Once selected, you will need to define the

Field name

and

Field Type

for the added values. You can also concat (join) two defined fields and create a custom field.

To create a custom fied, click on

Add Operation

, then select a

Field Value

from a dropdown, choose the CONCAT operator using the

Operator

dropdown and finally choose the second

Field Value

. Click on

Add Field

to add the custom field. Do remember you will need to define the

Field name

and

Field value

for the concated field.

Custom Plugin

â€‹

Custom Plugin parser comes into picture when you have highly-specific parsing needs. You can create a script-based parser that can analyze and parse the data tailored to your requirements. Write the script for your parser using the

Log Parser Plugin

screen. After creating a plugin, you can choose it using the

Select Plugin

dropdown.