



Recommendations for Logging and Monitoring

HITACHI

Inspire the Next

Change log (if you want to use it):

Date	Version	Author	Changes

Contents

- Overview..... 1
 - Before You Begin..... 1
 - Terms You Should Know 1
 - Other Prerequisites 1
 - Use Case: Monitoring System Resources 2
- Logging and Monitoring Best Practices..... 3
 - Logging Best Practices..... 3
 - Enable HTTP Logging..... 3
 - Enable Thread Logging..... 3
 - Enable Mondrian Logging..... 4
 - Enable Log Rotation..... 4
 - Track Logins and Failed Logins 4
 - Set Up Time-based Rolling Policy for Logs 4
 - Monitoring Best Practice..... 5
 - Turn Off Pentaho Operations Mart Logging 5
- Related Information 5
- Finalization Checklist..... 6

Overview

We have collected a series of best practice recommendations for logging and monitoring your Pentaho server environment. Some of the things discussed here include enabling HTTP, thread, and Mondrian logging, along with log rotation recommendations.

Software	Version(s)
Pentaho	6.x, 7.x, 8.0

The [Components Reference](#) in Pentaho Documentation has a complete list of supported software and hardware.

Before You Begin

Before beginning, use the following information to prepare for the procedures described in the main section of the document.

Terms You Should Know

Here are some terms you should be familiar with:

- **Network Logging** - the process of collecting, then storing, data over varying periods of time so that network trends can be analyzed.
- **Network Monitoring** - a network's effort to detect slow or failing components. These could include things such as failing routers, switches, or servers.
- **Network Auditing** - a process in which a network is analyzed for performance, security, availability, and management.

Other Prerequisites

This document assumes that you have knowledge about computer networking concepts and that you have already [installed and configured Pentaho](#). In the Pentaho Documentation, the [Monitor System Performance Using Pentaho Operations Mart](#) and the article called [Data Integration Operations Mart](#) have information about installing and configuring the Operations Marts.

More information about related topics outside of this document can be found in the [Pentaho Administration Guide](#) in the [Pentaho Documentation](#).

Use Case: Monitoring System Resources

Glorfindel must monitor systems resource usage for his new installation of Pentaho. This will help give him a bigger picture of how many resources are being used and the effect on network systems. He plans to use Operations Mart reports for this purpose, but wants to make sure that he has a good handle on different best practices for logging and monitoring with Pentaho before he gets started.

Logging and Monitoring Best Practices

Pentaho servers use the Apache log4j Java logging framework to store server feedback. The default settings in the `log4j.xml` file may be too verbose and grow too large for some production environments.

Logging Best Practices

This section contains recommended best practices to manage your logging operations for your Pentaho servers.

Enable HTTP Logging

- **Recommendation:** HTTP logging should be enabled.
- **Rationale:** CTools dashboards, report viewer, Dashboard Designer, and PUC make use of multiple HTTP requests to communicate with the BA server.
- **Solution:** Enabling HTTP logging will allow these and other external applications to be tracked at the request level. To enable HTTP logging, the `server.xml` file in `tomcat/conf` must be modified to have the appropriate entry. Insert the following line near the end of the file:

```
<Valve className="org.apache.catalina.valves.AccessLogValve"
directory="logs" prefix="http_access." suffix=".log"
pattern="%t |%a |%m |%p |%U |%q |%s |%S |%u |%D |%T |%B |%{User-
agent}i |%{Referer}i |%A |%{Content-Type}o" resolveHosts="false"/>
```

Enable Thread Logging

- **Recommendation:** Enable thread logging for better diagnosing or debugging.
- **Rationale:** The additional context of the thread that logged the item can help with diagnosing and debugging issues. Including this additional information does not add any significant overhead and should be considered when enabling logging for any category.
- **Solution:** Modify the `log4j.xml` file `ConversionPattern` parameter to have the `%t`. The full pattern includes the thread `%t` and is supplied in the `log4j.xml` file shipped with Pentaho. However, it is commented out. Simply uncomment the full pattern and comment out the default pattern.

Enable Mondrian Logging

- **Recommendation:** Enable Mondrian logging to log `mdx` and `sql` statements.
- **Rationale:** These settings are provided in the `log4j.xml` supplied by Pentaho, but are commented out by default. Enabling will cause logging to occur within the `log4j` categories `mondrian.mdx` and `mondrian.sql`. These categories log the statements and how long they take to execute. The SQL log also records the number of results returned in the result set. This insight can be extremely helpful in debugging and diagnosing query related issues.
- **Solution:** To enable Mondrian SQL and MDX logging, simply uncomment the sections of the `log4j.xml` related to `Mondrian.mdx` and `Mondrian.sql` categories.

Enable Log Rotation

- **Recommendation:** Use date-based or size-based log rotation with compression.
- **Rationale:** The default settings in the `log4j.xml` configuration file may be too verbose and cause the Pentaho log to grow too large for some production environments.
- **Solution:** These instructions for [Log Rotation](#) will help you setup either date-based or size-based log rotation with compression.

Track Logins and Failed Logins

- **Recommendation:** Enable logging for logins and failed logins.
- **Rationale:** Keeping track of failed logins on your servers helps you learn about unauthorized login attempts.
- **Solution:** If you want to keep a log for logins and failed logins, modify the `log4j.xml` file and add the following:

```
<category name="org.springframework.security.event.authentication">
<priority value="WARN"/>
</category>
```

Set Up Time-based Rolling Policy for Logs

- **Recommendation:** Implement a time-based rolling policy for your logs.
- **Rationale:** Using a time-based rolling policy for your logs will prevent them from growing too large for your production environment.
- **Solution:** If you want to set up a time-based rolling policy, modify the `log4j.xml` file and add the following:

```
<rollingPolicy class="org.apache.log4j.rolling.TimeBasedRollingPolicy">
  <param name="FileNamePattern"
value="/home/pentaho/pentaho/server/biserver-ee/pentaho.log_%d{yyyy-MM}.gz"/>
</rollingPolicy>
```

Monitoring Best Practice

Pentaho Operations Mart is the out-of-the-box monitoring tool and runs verbose logging by default.

Turn Off Pentaho Operations Mart Logging

- **Recommendation:** Turn off Pentaho Operations Mart logging.
- **Rationale:** The Pentaho Operations Mart ETL process runs multiple times a day and produces verbose messages which are unnecessary and could slow down a production environment.
- **Solution:** These messages can be turned off by altering the `xaction` logging level.

Related Information

Here are some links to information that you may find helpful while using this best practice document:

- Pentaho Documentation:
 - [Performance Monitoring and Logging](#)
 - [Log Rotation](#)
 - [Logging and Monitoring Operations with PDI](#)

Finalization Checklist

This checklist is designed to be added to any implemented project that uses this collection of best practices, to verify that all items have been considered and reviews have been performed.

Name of the Project: _____

Date of the Review: _____

Name of the Reviewer: _____

Item	Response	Comments
Did you enable:	YES_____ NO_____	
HTTP logging?	YES_____ NO_____	
thread logging?	YES_____ NO_____	
Mondrian logging?	YES_____ NO_____	
log rotation?	YES_____ NO_____	
Are you tracking logins and failed logins?	YES_____ NO_____	
Did you turn off Pentaho Operations Mart logging?	YES_____ NO_____	