rsyslog logging

This document explains the structure of Redis Enterprise Software log entries in rsyslog and how to use these log entries to identify events.

Note: You can also secure your logs with a remote logging server and log rotation.

Log concepts

Redis Enterprise Software logs information from a variety of components in response to actions and events that occur within the cluster.

In some cases, a single action, such as removing a node from the cluster, may actually consist of several events. These actions may generate multiple log entries.

All log entries displayed in the admin console are also written to syslog. You can configure rsyslog to monitor syslog. Enabled alerts are logged to syslog and appear with other log entries.

Types of log entries

Log entries are categorized into events and alerts. Both types of entries appear in the logs, but alert log entries also include a boolean “state” parameter that indicates whether the alert is enabled or disabled.

Log entries include information about the specific event that occurred. See the log entry tables for clusters, databases, nodes, and users for more details.

Severity

You can also configure rsyslog to add other information, such as the event severity.

Since rsyslog entries do not include severity by default, you can follow these steps to enable it:

1. Add the following line to /etc/rsyslog.conf:

   $template TraditionalFormatWithPRI,"%pri-text%: %timegenerated% %HOSTNAME% %syslogtag%msg:::drop-last-lf\n"

2. Modify $ActionFileDefaultTemplate to use your new template
   $ActionFileDefaultTemplateTraditionalFormatWithPRI

3. Save these changes and restart rsyslog to apply them

You can see the log entries for alerts and events in the /var/log/messages file.

Command components:

- %pri text adds the severity
- %timegenerated% adds the timestamp
- %HOSTNAME% adds the machine name
Log entry structure

The log entries have the following basic structure:

```
event_log[<process id>]:{<list of key-value pairs in any order>}
```

- **event_log**: Plain static text is always shown at the beginning of the entry.
- **process id**: The ID of the logging process
- **list of key-value pairs in any order**: A list of key-value pairs that describe the specific event. They can appear in any order. Some key-value pairs are always shown, and some appear depending on the specific event.
  - **Key-value pairs that always appear**:
    - "type": A unique code name for the logged event. For the list of codenames, see the logged events and alerts tables for clusters, databases, nodes, and users.
    - "object": Defines the object type and ID (if relevant) of the object this event relates to, such as cluster, node with ID, BDB with ID, etc. Has the format of `<object_type>[:<id>]`.
    - "time": Unix epoch time but can be ignored in this context.
  - **Key-value pairs that might appear depending on the specific entry**:
    - "state": A boolean where true means the alert is enabled, and false means the alert is disabled. This is only relevant for alert log entries.
    - "global_threshold": The value of a threshold for alerts related to cluster or node objects.
    - "threshold": The value of a threshold for alerts related to a BDB object.

Log entry samples

This section provides examples of log entries that include the rsyslog configuration to add the severity, timestamp, and machine name.

**Ephemeral storage passed threshold**

"Alert on" log entry sample

```
daemon.warning: Jun 14 14:49:20 node1 event_log[3464]:
{
    "storage_util": 90.061643120001,
    "global_threshold": "70",
    "object": "node:1",
    "state": true,
    "time": 1434282560,
    "type": "ephemeral_storage"
}
```

In this example, the storage utilization on node 1 reached the value of ~90%, which triggered the alert for "Ephemeral storage has reached 70% of its capacity."

Log entry components:

- daemon.warning - Severity of entry is warning
- Jun 14 14:49:20 - The timestamp of the event
* node1: Machine name
  * event_log: Static text that always appears
  * [3464]: Process ID
  * "storage_util":90.061643120001: Current ephemeral storage utilization
  * "global_threshold": "70": The user-configured threshold above which the alert is raised
  * "object": "node:1": The object related to this alert
  * "state": true: Current state of the alert
  * "time": 1434282560: Can be ignored
  * "type": "ephemeral_storage": The code name of this specific event. See logged node alerts and events for more details.

"Alert off" log entry sample

```
demon.info: Jun 14 14:51:35 node1 event_log[3464]:
{
  "storage_util":60.051723520008,
  "global_threshold": "70",
  "object": "node:1",
  "state":false,
  "time": 1434283480,
  "type": "ephemeral_storage"
}
```

This log entry is an example of when the alert for the node with ID 1 "Ephemeral storage has reached 70% of its capacity" has been turned off as result of storage utilization reaching the value of ~60%.

Log entry components:
  * daemon.info: Severity of entry is info
  * Jun 14 14:51:35: The timestamp of the event
  * node1: Machine name
  * event_log: Static text that always appears
  * [3464]: Process ID
  * "storage_util": 60.051723520008: Current ephemeral storage utilization
  * "global_threshold": "70": The user configured threshold above which the alert is raised (70% in this case)
  * "object": "node:1": The object related to this alert
  * "state": false: Current state of the alert
  * "time": 1434283480: Can be ignored
  * "type": "ephemeral_storage": The code name identifier of this specific event. See logged node alerts and events for more details.

Odd number of nodes with a minimum of three nodes alert
"Alert on" log entry sample

daemon.warning: Jun 14 15:25:00 node1 event_log[8310]:
{
   "object":"cluster",
   "state": true,
   "time": 1434284700,
   "node_count": 1,
   "type":"even_node_count"
}

This log entry is an example of when the alert for "True high availability requires an odd number of nodes with a minimum of three nodes" has been turned on as result of the cluster having only one node.

Log entry components:

- daemon.warning - Severity of entry is warning
- Jun 14 15:25:00 - The timestamp of the event
- node1 - Machine name
- event_log - Static text that always appears
- [8310] - Process ID
- "object":"cluster" - The object related to this alert
- "state":true - Current state of the alert
- "time":1434284700 - Can be ignored
- "node_count":1 - The number of nodes in the cluster
- "type":"even_node_count" - The code name identifier of this specific event. See logged cluster alerts and events for more details.

"Alert off" log entry sample

daemon.warning: Jun 14 15:30:40 node1 event_log[8310]:
{
   "object":"cluster",
   "state": false,
   "time": 1434285200,
   "node_count": 3,
   "type":"even_node_count"
}

This log entry is an example of when the alert for "True high availability requires an odd number of nodes with a minimum of three nodes" has been turned off as result of the cluster having 3 nodes.

Log entry components:

- daemon.warning - Severity of entry is warning
- Jun 14 15:30:40 - The timestamp of the event
- node1 - Machine name
Node has insufficient disk space for AOF rewrite

"Alert on" log entry sample

```
daemon.err: Jun 15 13:51:23 node1 event_log[34252]:
{
    "used": 23457188,
    "missing": 604602126,
    "object": "node:1",
    "free": 9867264,
    "needed": 637926578,
    "state": true,
    "time": 1434365483,
    "disk": 705667072,
    "type": "insufficient_disk_aofrw"
}
```

This log entry is an example of when the alert for "Node has insufficient disk space for AOF rewrite" has been turned on as result of not having enough persistent storage disk space for AOF rewrite purposes. It is missing 604602126 bytes.

Log entry components:

- `daemon.err` - Severity of entry is error
- `Jun 15 13:51:23` - The timestamp of the event
- `node1` - Machine name
- `event_log` - Static text that always appears
- `[34252]` - Process ID
- "used":23457188 - The amount of disk space in bytes currently used for AOF files
- "missing":604602126 - The amount of disk space in bytes that is currently missing for AOF rewrite purposes
- "object":"node:1" - The object related to this alert
- "free":9867264 - The amount of disk space in bytes that is currently free
- "needed":637926578 - The amount of total disk space in bytes that is needed for AOF rewrite purposes
- "state":true - Current state of the alert
- "time":1434365483 - Can be ignored
"disk": 705667072 - The total size in bytes of the persistent storage

"type": "insufficient_disk_aofrw" - The code name of this specific event. See logged node alerts and events for more details.

"Alert off" log entry sample

daemon.info: Jun 15 13:51:11 node1 event_log[34252]:
{
    "used": 0, "missing":-21614592,
    "object": "node:1",
    "free": 21614592,
    "needed": 0,
    "state":false,
    "time": 1434365471,
    "disk": 705667072,
    "type": "insufficient_disk_aofrw"
}

Log entry components:

- `daemon.info` - Severity of entry is info
- `Jun 15 13:51:11` - The timestamp of the event
- `node1` - Machine name
- `event_log` - Static text that always appears
- `[34252]` - Process ID
- "used": 0 - The amount of disk space in bytes currently used for AOF files
- "missing": -21614592 - The amount of disk space in bytes that is currently missing for AOF rewrite purposes. In this case, it is not missing because the number is negative.
- "object": "node:1" - The object related to this alert
- "free": 21614592 - The amount of disk space in bytes that is currently free
- "needed": 0 - The amount of total disk space in bytes that is needed for AOF rewrite purposes. In this case, no space is needed.
- "state":false - Current state of the alert
- "time": 1434365471 - Can be ignored
- "disk": 705667072 - The total size in bytes of the persistent storage
- "type": "insufficient_disk_aofrw" - The code name of this specific event. See logged node alerts and events for more details.