

Winston logging to OpenObserve

Centralized logging for Node.js apps.

1. Install OpenObserve. Installation is simply downloading and unzipping the single binary. OpenObserve runs externally by default with no option to change this behavior, so you may need a firewall to restrict the access. I used ufw.
2. Create a 'data' directory in the same location as the OpenObserve binary. Start OpenObserve:

```
ZO_ROOT_USER_EMAIL="<<email>" ZO_ROOT_USER_PASSWORD="<<password>"  
ZO_DATA_DIR="$PWD/data" ./openobserve > /dev/null 2>&1 &
```

3. Add winston as a dependency to your Node.js app.
4. Create logger.js:

```
import winston, { transports } from "winston";  
  
const options = {  
  level: 'debug',  
  host: 'localhost',  
  port: '5080',  
  path: '/api/<organization>/<stream>/_json',  
  auth: {  
    username: '<email>',  
    password: '<password>',  
  },  
  ssl: false,  
  batch: true,  
  batchInterval: 5000,  
  batchCount: 10,  
};  
  
export const log = winston.createLogger({  
  transports: new transports.Http(options),  
  exceptionHandlers: new transports.Http(options),
```

```
    exitOnError: false
  })
```

5. Import and use the logger in the application:

```
import {log} from "$lib/logger.js";

export async function handle({ event, resolve }) {
  log.debug(event.request.method + ' ' + event.url.pathname + event.url.search);
  ...
}
```

6. Run the application and view the logs in OpenObserve:

The screenshot shows the OpenObserve web interface. At the top, there's a navigation bar with the OpenObserve logo, 'OpenAPI', 'Docs', language settings (English), user 'orryworx', and a profile icon for 'admin@orryworx.com'. Below the navigation bar, there are tabs for 'Histogram', 'SQL Mode', and 'Syntax Guide'. A search bar and a 'Run query' button are also visible. The main content area is split into two panes: 'Query Editor' and 'VRL Function Editor'. The 'Histogram' view is active, showing a bar chart with the x-axis representing time (from 10:55:30 to 10:59:00) and the y-axis representing the number of hits (0 to 40). A tooltip indicates 'Showing 122 out of 122 hits in 21 ms. (Scan Size: 0MB)'. Below the histogram, a table of log entries is displayed, with columns for 'timestamp' and 'source'. The log entries are filtered to show only 'debug' level messages. The messages include various API calls and database queries, such as 'Privilege.findMany 3ms', 'Account.findMany 2ms', and several 'Query' messages for SELECT and GET requests.

timestamp	source
> Jul 06, 2023 10:59:23.903 +05:30	{"_timestamp":1688621363903746,"level":"debug","message":"Privilege.findMany 3ms"}
> Jul 06, 2023 10:59:23.903 +05:30	{"_timestamp":1688621363903727,"level":"debug","message":"Params: [{"clft7bemj0006cplvllvaex1b","clft7bl0p0008cplvjh3i81b","clft7bq..."}]}
> Jul 06, 2023 10:59:23.903 +05:30	{"_timestamp":1688621363903719,"level":"debug","message":"Query: SELECT `main`.`Permission`.`id`, `main`.`Permission`.`role`, `main`.`Pe...`"}
> Jul 06, 2023 10:59:23.903 +05:30	{"_timestamp":1688621363903703,"level":"debug","message":"Params: [{"clgs2qneq0014y8u8xvnnwess",-1,0}]"}
> Jul 06, 2023 10:59:23.903 +05:30	{"_timestamp":1688621363903693,"level":"debug","message":"Query: SELECT `main`.`Privilege`.`id`, `main`.`Privilege`.`merchant_id`, `main...`"}
> Jul 06, 2023 10:59:23.903 +05:30	{"_timestamp":1688621363903684,"level":"debug","message":"GET /configuration/clgs2qneq0014y8u8xvnnwess"}
> Jul 06, 2023 10:59:23.903 +05:30	{"_timestamp":1688621363903674,"level":"debug","message":"Account.findMany 2ms"}
> Jul 06, 2023 10:59:23.903 +05:30	{"_timestamp":1688621363903665,"level":"debug","message":"Params: [{"clgs2qneq0014y8u8xvnnwess",false,-1,0}]"}
> Jul 06, 2023 10:59:23.903 +05:30	{"_timestamp":1688621363903652,"level":"debug","message":"Query: SELECT `main`.`Account`.`id`, `main`.`Account`.`bank_routing_number`, `r...`"}
> Jul 06, 2023 10:59:23.903 +05:30	{"_timestamp":1688621363903614,"level":"debug","message":"Privilege.findMany 4ms"}
> Jul 06, 2023 10:59:23.751 +05:30	{"_timestamp":1688621363751209,"level":"debug","message":"Params: [{"clft7bemj0006cplvllvaex1b","clft7bl0p0008cplvjh3i81b","clft7bq..."}]}
> Jul 06, 2023 10:59:23.751 +05:30	{"_timestamp":1688621363751193,"level":"debug","message":"Query: SELECT `main`.`Permission`.`id`, `main`.`Permission`.`role`, `main`.`Pe...`"}
> Jul 06, 2023 10:59:23.751 +05:30	{"_timestamp":1688621363751161,"level":"debug","message":"Params: [{"clgs2qneq0014y8u8xvnnwess",-1,0}]"}
> Jul 06, 2023 10:59:23.751 +05:30	{"_timestamp":1688621363751137,"level":"debug","message":"Query: SELECT `main`.`Privilege`.`id`, `main`.`Privilege`.`merchant_id`, `main...`"}
> Jul 06, 2023 10:59:23.751 +05:30	{"_timestamp":1688621363751124,"level":"debug","message":"GET /accounts?merchant_id=clgs2qneq0014y8u8xvnnwess"}
> Jul 06, 2023 10:59:23.751 +05:30	{"_timestamp":1688621363751110,"level":"debug","message":"Privilege.findMany 2ms"}

Revision #1

Created 6 July 2023 04:59:29 by Admin

Updated 6 July 2023 06:19:54 by Admin