

Tutorial 1: Logging the Transaction of a Simple HTTP Server

This first tutorial covers the basics of logging the transactions of a simple HTTP server. It shows how easy it is to start using the E2E Transaction Logger to log your transactions in a structured way. This allows you to get log files which can be analyzed by hand but preferably using the [Scheer E2E Dashboards](#) that generate graphical views of how your services perform.

Building a simple example

Creating the server

Let's use a simple example taken directly from the [Node.js homepage](#).

```
var http = require('http');
http.createServer(function (req, res) {
  res.writeHead(200, {'Content-Type': 'text/plain'});
  res.end('Hello World\n');
}).listen(1337, '127.0.0.1');
console.log('Server running at http://127.0.0.1:1337/');
```

Starting this and going to <http://127.0.0.1:1337/> with your favorite browser will simply show "Hello world".

Now we have a simple HTTP server and we can make requests to it.

Logging the requests

Let's start using the E2E Transaction Logger to trace our requests. First add a dependency to **e2e-transaction-logger** in your package.json and use it to trace the hello world requests.

```
var http = require('http');
var e2eLogger = require('e2e-transaction-logger');
http.createServer(function (req, res) {
  var trx = e2eLogger.startTransaction('Hello World');
  res.writeHead(200, {'Content-Type': 'text/plain'});
  res.end('Hello World\n');
  trx.end();
}).listen(1337, '127.0.0.1');
console.log('Server running at http://127.0.0.1:1337/');
```

We use **startTransaction** at the beginning to create the transaction and simply call **end** on the returned object when it is done.

If you run again your HTTP server and do some hello world requests you will see that a folder logs is created with a file called **transaction_YYYY-MM-DD.log**.

It contains info about the transactions you made, e.g.:

On this Page:

- [Building a simple example](#)
 - [Creating the server](#)
 - [Logging the requests](#)
 - [Transaction log files](#)
- [Extending the Logging](#)
 - [Multiple transactions](#)
 - [Success or fail](#)
- [Dashboards](#)

Related Pages:

- [Documentation of the E2E Transaction Logger](#)
- [Tutorial 1: Logging the Transactions of a Simple HTTP Server](#)
- [Tutorial 2: Backend Access](#)
- [Tutorial 3: Express](#)
- [Scheer E2E Dashboards](#)

2014-04-09	15:22:27	+0200	43aa5924-1403-4102-a9d1-
0c861ab9cb59	1	Hello World	0 OK
INTERFACE	SERVICE_ENTER		
2014-04-09	15:22:27	+0200	43aa5924-1403-4102-a9d1-
0c861ab9cb59	1	Hello World	5 OK
INTERFACE	SERVICE_EXIT		
2014-04-09	15:22:27	+0200	73fb299c-ffcb-4e01-831e-
4b481e5b003b	2	Hello World	0 OK
INTERFACE	SERVICE_ENTER		
2014-04-09	15:22:27	+0200	73fb299c-ffcb-4e01-831e-
4b481e5b003b	2	Hello World	2 OK
INTERFACE	SERVICE_EXIT		
2014-04-09	15:22:59	+0200	f1920118-ec33-4da9-b7f7-
43eeaf9872d2	3	Hello World	0 OK
INTERFACE	SERVICE_ENTER		
2014-04-09	15:22:59	+0200	f1920118-ec33-4da9-b7f7-
43eeaf9872d2	3	Hello World	1 OK
INTERFACE	SERVICE_EXIT		
2014-04-09	15:22:59	+0200	7aa73586-b368-4b3e-bea0-
352d5eca63c1	4	Hello World	0 OK
INTERFACE	SERVICE_ENTER		
2014-04-09	15:22:59	+0200	7aa73586-b368-4b3e-bea0-
352d5eca63c1	4	Hello World	1 OK
INTERFACE	SERVICE_EXIT		

Transaction log files

In the transaction log files, each line corresponds to a log entry and the logged parameters are separated by tabs ('\t').

The parameters are in the following order:

Parameter	Format	Description
Date	YYYY-MM-DD	The current date.
Time	HH:mm:s	The time.
Timezone	+/-HHMM	The timezone offset.
Transaction ID	string	The transaction id is defined by the developer or uniquely generated.
Request ID	number	This is the unique id of the request on this server. It is automatically generated.
Name	string	The name of this transaction, defined by the developer.
Elapsed time	number	The number of milliseconds elapsed since the start of the transaction.
State	string	The state of the entry. Can be 'OK' or 'ERROR'. It is used to differentiate successful operations and failed ones.
Domain	string	The domain indicates which kind of entry it is. In case of transactions the domain is 'INTERFACE'.
Log type	string	The log type indicates the type of entry. In case of transactions it can be 'SERVICE_ENTER' on transaction start and 'SERVICE_EXIT' on transaction end.
Parameter 1		The parameters are used to trace particular data of the entry. In case of transactions there are none.
Parameter 2		Same as parameter 1.

Extending the Logging

Multiple transactions

On a real server, you would have different transactions. This is when logging becomes interesting to see the count of of each, the time they took and more.

```
http.createServer(function (req, res) {
  var trx;
  switch(req.url){
    case '/hello':
      trx = e2eLogger.startTransaction('Say Hello')
      sayHello(req, res, function(){
        trx.end();
      });
      break;
    case '/goodbye':
      trx = e2eLogger.startTransaction('Say Good Bye')
      sayGoodBye(req, res, function(){
        trx.end();
      });
      break;
    case '/do-something':
      trx = e2eLogger.startTransaction('Do Something')
      doSomething(req, res, function(){
        trx.end();
      });
      break;
    default:
      res.end();
  }
}).listen(1337, '127.0.0.1');
```

And you will then get some logs looking like this:

2014-04-09	15:59:41	+0200	e8275fb5-549a-415f-b905-
b4014437123b	1	Say Hello	0 OK
INTERFACE	SERVICE_ENTER		
2014-04-09	15:59:41	+0200	e8275fb5-549a-415f-b905-
b4014437123b	1	Say Hello	3 OK
INTERFACE	SERVICE_EXIT		
2014-04-09	15:59:48	+0200	4235ecb3-fab7-45ab-ac97-
de53d01fd6b8	2	Do Something	0 OK
INTERFACE	SERVICE_ENTER		
2014-04-09	15:59:49	+0200	4235ecb3-fab7-45ab-ac97-
de53d01fd6b8	2	Do Something	823 OK
INTERFACE	SERVICE_EXIT		
2014-04-09	15:59:56	+0200	74961668-363c-43cb-aae8-
33a5381fefdb	3	Say Hello	0 OK
INTERFACE	SERVICE_ENTER		
2014-04-09	15:59:56	+0200	74961668-363c-43cb-aae8-
33a5381fefdb	3	Say Hello	1 OK
INTERFACE	SERVICE_EXIT		
2014-04-09	15:59:57	+0200	41ea397d-4c10-4bd0-9e27-
9540159e85bd	4	Say Hello	0 OK
INTERFACE	SERVICE_ENTER		
2014-04-09	15:59:57	+0200	41ea397d-4c10-4bd0-9e27-
9540159e85bd	4	Say Hello	2 OK
INTERFACE	SERVICE_EXIT		
2014-04-09	16:00:07	+0200	173e7bd5-29ce-4d88-a5ae-
8b83a3b16a7c	5	Do Something	0 OK
INTERFACE	SERVICE_ENTER		
2014-04-09	16:00:08	+0200	173e7bd5-29ce-4d88-a5ae-
8b83a3b16a7c	5	Do Something	497 OK
INTERFACE	SERVICE_EXIT		
2014-04-09	16:00:31	+0200	84e3469a-eaf6-43c5-bd04-
ffb14906019	6	Say Good Bye	0 OK
INTERFACE	SERVICE_ENTER		
2014-04-09	16:00:31	+0200	84e3469a-eaf6-43c5-bd04-
ffb14906019	6	Say Good Bye	1 OK
INTERFACE	SERVICE_EXIT		

Success or fail

The end function of a transaction takes a parameter which allow you to set the success or fail of the transaction. The valid values are true or 'OK' for success and false or 'ERROR' for fail.

```
case '/do-something':
  trx = e2eLogger.startTransaction('Do Something')
  doSomething(req, res, function(success){
    trx.end(success);
  });
  break;
```

Then you can see the state of the end entries.

```

2014-04-09 16:14:32 +0200 221bb52f-a96c-4858-8af3-65cca3258214
10 Say Hello 0 OK INTERFACE SERVICE_ENTER
2014-04-09 16:14:32 +0200 221bb52f-a96c-4858-8af3-65cca3258214
10 Say Hello 1 OK INTERFACE SERVICE_EXIT
2014-04-09 16:14:36 +0200 41e1b8e1-0728-4971-b0e5-ae9778f680fd
11 Do Something 0 OK INTERFACE SERVICE_ENTER
2014-04-09 16:14:36 +0200 41e1b8e1-0728-4971-b0e5-ae9778f680fd
11 Do Something 790 OK INTERFACE SERVICE_EXIT
2014-04-09 16:14:38 +0200 bf5e1563-a2b2-4da7-8d51-b30beb55fae8
12 Do Something 0 OK INTERFACE SERVICE_ENTER
2014-04-09 16:14:38 +0200 bf5e1563-a2b2-4da7-8d51-b30beb55fae8
12 Do Something 303 ERROR INTERFACE SERVICE_EXIT
2014-04-09 16:14:41 +0200 07d52997-80fa-4bba-82ee-c4588618da95
13 Say Hello 0 OK INTERFACE SERVICE_ENTER
2014-04-09 16:14:41 +0200 07d52997-80fa-4bba-82ee-c4588618da95
13 Say Hello 1 OK INTERFACE SERVICE_EXIT
2014-04-09 16:14:44 +0200 8a0a47b1-0b32-45f6-888a-7ec2aa65e0e7
14 Do Something 0 OK INTERFACE SERVICE_ENTER
2014-04-09 16:14:45 +0200 8a0a47b1-0b32-45f6-888a-7ec2aa65e0e7
14 Do Something 195 OK INTERFACE SERVICE_EXIT
2014-04-09 16:14:50 +0200 e3098583-8b08-475b-87fd-c326d3bfed0c
15 Say Good Bye 0 OK INTERFACE SERVICE_ENTER
2014-04-09 16:14:50 +0200 e3098583-8b08-475b-87fd-c326d3bfed0c
15 Say Good Bye 1 OK INTERFACE SERVICE_EXIT
2014-04-09 16:14:53 +0200 10a8b04d-f753-4a30-b5bc-6b7597ff310a
16 Do Something 0 OK INTERFACE SERVICE_ENTER
2014-04-09 16:14:54 +0200 10a8b04d-f753-4a30-b5bc-6b7597ff310a
16 Do Something 798 ERROR INTERFACE SERVICE_EXIT
2014-04-09 16:14:56 +0200 7c7f580b-72f5-43bb-b8ac-20f991f105af
17 Do Something 0 OK INTERFACE SERVICE_ENTER
2014-04-09 16:14:56 +0200 7c7f580b-72f5-43bb-b8ac-20f991f105af
17 Do Something 30 ERROR INTERFACE SERVICE_EXIT
2014-04-09 16:14:56 +0200 981d75be-178e-4387-96e0-158a3e881af6
18 Do Something 0 OK INTERFACE SERVICE_ENTER
2014-04-09 16:14:57 +0200 981d75be-178e-4387-96e0-158a3e881af6
18 Do Something 610 OK INTERFACE SERVICE_EXIT
2014-04-09 16:14:59 +0200 b5db172e-9add-4c10-9ef6-803b258ea968
19 Do Something 0 OK INTERFACE SERVICE_ENTER
2014-04-09 16:14:59 +0200 b5db172e-9add-4c10-9ef6-803b258ea968
19 Do Something 229 ERROR INTERFACE SERVICE_EXIT
2014-04-09 16:15:04 +0200 ec7d0643-33fd-4001-b834-e034486d0f64
20 Say Good Bye 0 OK INTERFACE SERVICE_ENTER
2014-04-09 16:15:04 +0200 ec7d0643-33fd-4001-b834-e034486d0f64
20 Say Good Bye 1 OK INTERFACE SERVICE_EXIT

```

You can download the complete example file from this page ([httpServer.js](http://Server.js)).

Dashboards

These log files are fully compatible with the [E2E Dashboards](#) services, so they can be loaded into the database and used with the dashboards to analyze the performance of your services.

Find below some examples of the views you would get with this simple HTTP server.

Figure: Table View of the E2E Service Dashboard

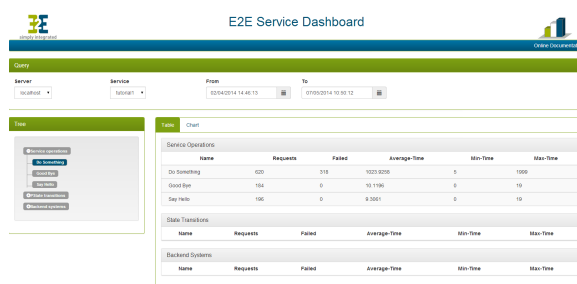


Figure: Graphical View of the E2E Service Dashboard

