

AWS Elastic Beanstalk Import Integration

By combining AWS Elastic Beanstalk and Treasure Data, you can achieve scalability in both your front-end and back-end systems immediately.

- [Prerequisites](#)
- [Installing Treasure Agent](#)
- [Importing Data from Applications](#)

Prerequisites

- Basic knowledge of Treasure Data
- Basic knowledge of AWS Elastic Beanstalk

Installing Treasure Agent

Follow the instructions below to install Treasure Agent (td-agent) our streaming data collector, into your application.

What's td-agent?

Treasure Data's td-agent is a distributed log collector that is required for use of Treasure Data on your AWS Elastic Beanstalk container. You use td-agent to import logs from your AWS Elastic Beanstalk application to Treasure Data's cloud data warehouse.

Installation

A boilerplate application to install td-agent within your container is available in the following repository:

- [treasure-data/elastic-beanstalk-td-agent](https://github.com/treasure-data/elastic-beanstalk-td-agent)

Install .ebextensions Scripts

Download two files from github as shown in the following example and put the files in your .ebextensions directory.

```
$ cd $YOUR_REPO
$ mkdir -p .ebextensions
$ wget -O .ebextensions/0-td-agent-gen-config.config 'https://raw.githubusercontent.com/treasure-data/elastic-beanstalk-td-agent/master/.ebextensions/0-td-agent-gen-config.config'
$ wget -O .ebextensions/1-td-agent-install.config 'https://raw.githubusercontent.com/treasure-data/elastic-beanstalk-td-agent/master/.ebextensions/1-td-agent-install.config'
```

Set Your API Key

Next, put your API key in the appropriate location within your configuration file.

```
$ td apikey:show
$ emacs .ebextensions/0-td-agent-gen-config.config
```

Deploy

Finally, deploy the changes into production.

```
$ git add .ebextensions/0-td-agent-gen-config.config
$ git add .ebextensions/1-td-agent-install.config
$ git commit -a -m 'custom extensions to install td-agent'
$ git aws.push
```

Confirm Data Upload

You can import logs from your local system into Treasure Data by issuing curl commands.

```
$ curl -X POST -d 'json={"action":"login","user":2}' http://localhost:8888/td.testdb.www_access
$ curl -X POST -d 'json={"action":"login","user":2}' http://localhost:8888/td.testdb.www_access
$ curl -X POST -d 'json={"action":"login","user":2}' http://localhost:8888/td.testdb.www_access
$ curl -X POST -d 'json={"action":"login","user":2}' http://localhost:8888/td.testdb.www_access
$ curl -X POST -d 'json={"action":"login","user":2}' http://localhost:8888/td.testdb.www_access
```

After a few minutes (usually about 5 minutes), you can issue the following command to confirm that your data has been uploaded into the cloud. If the count is greater than zero, the data upload succeeded.

```
$ td tables testdb
+-----+-----+-----+-----+
| Database | Table | Type | Count | Schema |
+-----+-----+-----+-----+
| testdb | www_access | log | 5 | |
+-----+-----+-----+-----+
```

Importing Data from Applications

You're now ready to import data from your applications to Treasure Data via td-agent. Choose the appropriate logger for your application's language.

Supported Languages		
Ruby or Rails	Java	Perl
Python	PHP	Scala
Node.js		