

Analytics > Log & Crash Search > Log4J v2 SDK Guide

Log & Crash Log4J v2 SDK sends logs to a Log & Crash Search collector server. Below describe benefits and features of Log & Crash Log4J v2 SDK.

- Send logs to a collector server.
- Retrieve and search logs sent from Log & Crash Search.
- Operate under a multi-threading environment.

Supporting Environment

- Log4J 2.x

Download

Go to [TOAST Document](#) and download **Log4J 2 SDK**.[\[DOCUMENTS\]](#) >

Click [Download] > [Analytics > Log & Crash Search] > [Log4J.v2 SDK]

Install

Configuration

SDK is configured as below.

```
docs/      ; Log4J 2 SDK Document
lib/       ; Log4J 2 Library
sample/    ; Log4J 2 Sample
```

SDK Sample

Below describe sample/log4j2/ that is provided.

1. Run Eclipse, execute **File > Import > Maven > Existing Maven Projects** in the menu and import sample/log4j2/.
2. Open the src/test/resources/log4j2.xml file and update with issued Appkey and version, and collector server address, if necessary.

```
<collectorUrl>https://api-logncrash.cloud.toast.com </collectorUrl>
<appkey>__app_key__</appkey>
<version>1.0.0</version>
```

3. Go to **Project > Properties > Java Build Path > Libraries** in Eclipse and add toast-logncrash-log4j2-sdk-jar.

4. In Eclipse, select **Run > Run As > JUnit Test** and execute.

Example

1. Add Log4j 2 SDK library to your project. For instance, select **Project > Properties > Java Build Path > Libraries** in the Eclipse menu and add toast-logncrash-log4j2-sdk.jar
2. In the case of Maven, add dependency to pom.xml.

```
<dependency>
  <groupId>org.apache.logging.log4j</groupId>
  <artifactId>log4j-api</artifactId>
  <version>2.3</version>
</dependency>
<dependency>
  <groupId>org.apache.logging.log4j</groupId>
  <artifactId>log4j-core</artifactId>
  <version>2.3</version>
</dependency>
<dependency>
  <groupId>commons-lang</groupId>
  <artifactId>commons-lang</artifactId>
  <version>2.5</version>
</dependency>
<dependency>
  <groupId>org.apache.httpcomponents</groupId>
  <artifactId>httpclient</artifactId>
  <version>4.2.6</version>
</dependency>
<dependency>
  <groupId>org.apache.httpcomponents</groupId>
  <artifactId>httpcore</artifactId>
  <version>4.2.4</version>
</dependency>
<dependency>
  <groupId>javax.servlet</groupId>
  <artifactId>servlet-api</artifactId>
  <version>2.4</version>
</dependency>
<dependency>
  <groupId>org.json</groupId>
  <artifactId>json</artifactId>
  <version>20090211</version>
</dependency>
```

For SLF4J, add the following dependency.

```

<dependency>
  <groupId>org.slf4j</groupId>
  <artifactId>slf4j-api</artifactId>
  <version>1.7.2</version>
</dependency>
<dependency>
  <groupId>org.apache.logging.log4j</groupId>
  <artifactId>log4j-slf4j-impl</artifactId>
  <version>2.3</version>
</dependency>

```

3. For non-Maven users, download libraries as below and add to the class path.

```

log4j, 2.3
log4j-core, 2.3
commons-lang, 2.5
httpclient, 4.2.6
httpcore, 4.2.4
servlet-api, 2.4
json, 20090211

```

4. For the setting and configuration of appender, write log4j2.xml.

- Refer to sample/log4j2/src/test/resources/log4j2.xml for the entire configuration.
- Make sure to use issued Appkey and collector server address, to appKey and collectorUrl, respectively.

```

<Appenders>
  <LogNcrashHttpAppender name="HTTP">
    <collectorUrl>https://api-logncrash.cloud.toast.com </collectorUrl>

    <appKey>__app_key__</appKey>
    <version>1.0.0</version>
    <logSource>http-log4j2</logSource>
    <logType>log</logType>
    <enable>true</enable>
    <debug>false</debug>
  </LogNcrashHttpAppender>
</Appenders>

...
<Loggers>
  <Root level="debug">
    <AppenderRef ref="HTTP"/>
  </Root>
</Loggers>

```

5. For Java, use as follows.

```
...
private static Logger log4j2Logger =
LogManager.getLogger(LogNCrashLog4j2Sample.class);
...
// CustomField Test
ThreadContext.put("custom", "custom log test");
log4j2Logger.debug("LogNCrash Log4J2 Test");
...
try {
    String npe = null;
    npe.toString();
} catch (Exception e) {
    log4j2Logger.error(e.toString(), e);
}
```

API List

Setting Items for log4j2.xml

- collectorUrl: Collector server address HTTP : <https://api-logncrash.cloud.toast.com>
- appKey: Project Appkey: required
- version: Project version. Default is "1.0.0".
- logSource: Log source. Default is "http-log4j2".
- logType: Log type. Default is "log".
- enable: Whether to use Appender or not. Default is "true".
- debug: Whether to use Debug or not. Default is "false".

Constraints

- **log4j 1.2** is not supported.

FAQ

How can I use Asynchronous Logger to enhance performance?

Make a reference of Asynchronous Loggers for Low-Latency Logging

How can a Java stack trace be logged to a log4j 2 (including Log & Crash Search)?

To get an output of stack trace with log4j 2, use the `log.error(e.toString(),e);` type: cannot get an output of stack trace for `log.error(e);`.

```
String[] aa= null;
try {
    aa[0] = "111";
} catch(NullPointerException e) {
    log.error(e); // no output of stacktrace
    log.error(e.toString(), e); // output of stacktrace
}
```