

```
agent.sources.httpSrc.handler.insertTimestamp = true
```

```
# Initializes a memory channel with default configuration
// 使用默认配置初始化 memory channel
agent.channels.memChannel.type = memory
```

What parameters the handler requires and how they are used by the handler depends on the specific handler implementation. Please consult the handler's documentation for details on this.

处理程序需要什么参数，以及这些参数是如何被处理程序使用的，取决于特定的处理程序实现。详情请查阅处理程序的文档。

针对 HTTP Source 写处理程序 *

对用户来说，开发和嵌入处理程序（handler）来转换接收到的数据为 Flume 事件是很简单的。HTTP Source 可以接收从客户端发来的能被处理程序处理的任何格式的数据。HTTP Source 处理程序是一个继承自简单接口 `HTTPSourceHandler` 的类：

```
package org.apache.flume.source.http;

public interface HTTPSourceHandler extends Configurable {
    public List<Event> getEvents(HttpServletRequest request) throws
        HTTPBadRequestException, Exception;
}
```

45

处理程序的接口很简单，只有一个方法，`getEvents`，该方法能接收客户端发送的 `HttpServletRequest`，并且返回 Flume 事件的列表。尽管处理程序接口很简单，但它基本能将 `HttpServletRequest` 中的输入数据做任意处理从而转化成 Flume 事件。但是处理的总量应该有所限制，否则客户端给 Source 发送数据可能导致超时。处理程序通过 Flume 的标准配置机制进行配置。因为不管是什么处理程序，HTTP Source 都是一个请求，使用恰好一个事务，发送方必须谨慎只发送管道能支持的尽量多的事件。

处理程序负责确保传递给它的配置参数是有效的。HTTP Source 将在启动时实例化和配置处理程序。因为 HTTP Source 将处理程序抛出的异常传递给配置系统，所以处理程序必须校验参数并且在 `configure` 方法中应用该参数。通过上下文实例将参数传递给 `configure` 方法。Context 实例是简单的键值对，包含各种配置键和它们的值。如果传来的配置是不合法的或不能被成功应用，HTTP Source 将再次通过处理程序将异常抛给配置系统，这会反过来禁用 HTTP Source 并将它从 Agent 中移除。

当处理传入的数据时，HTTP Source 通过返回客户端失败状态来操作处理程序抛出的异