1. 
2. **Agenda**:
   1. We will study the input plugin HTTP to send **events** to **logstash pipeline**.
3. Logically, the **plugin** is named as **http**.
4. There are also plugins for dealing with **TCP and UDP Sockets**.  
   But we will focus on communicating with **Logstash** through **HTTP** in this lecture.
5. A screenshot of a computer

   Description automatically generated with medium confidence  
   Let’s define configuration explicitly.   
   A screenshot of a computer

   Description automatically generated
6. Let’s send HTTP Request no matter which HTTP Client we use as logstash doesn’t care about it.
7. We will use two.
   1. cURL &
   2. POSTMAN : Graphical Tool.
8. A screenshot of a computer

   Description automatically generated with medium confidence
   1. We will use **PUT** request to send **Event Data**.
   2. **Event Data** is sent within the **Request Body**.
   3. We will send event as JSON.

But see we didn’t define the Codec for the HTTP Input Plugin.  
This is because, http input plugin will parse the received request body based on its Content-Type.  
As we will pass header **“Content-Type” : “application/json”**.  
We can configure this behaviour for other content type as well by using the option named **additional\_codecs.**

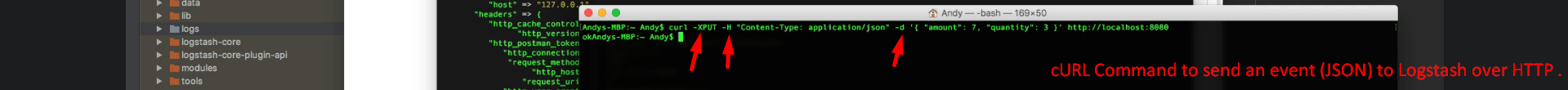
1. A screen shot of a computer

   Description automatically generated with low confidence
   1. Here we can see that the event that we just sent was received and processed by logstash.
   2. Within the process events in the console, we can see a field named **headers** apart from the fields we saw in earlier lectures.
   3. This headers object contains **HTTP Headers that Logstash** received.
      1. This object also contains “request\_path” => “/”.  
         In this example, we didn’t specify anything after the port in the URL but we could have just as well have done that.
      2. The request path doesn’t make any difference for Logstash but can be used in conditional statements and other processing.  
         **Example** could be to send business orders to “/business” and consumer orders to “/consumer” and then handle two kinds of orders differently.
      3. Did you notice that Logstash has no problem to handle **multiline json** with the request body?  
         Unlike we had entered the JSON within the terminal.  
         That is because **HTTP input plugin** will consider each HTTP request as **a single event** unless we send JSON array.

This is unlike other **plugins** which consider each line to be an event **by default**.

* + 1. Anyway, we have just successfully sent **an event to Logstash over HTTP.**  
       This is often very useful way of sending events because we will typically want to transfer **events over the network**.

1. Let’s see how to accomplish the same thing using **cURL Command**.
   1. **Sending event over HTTP using cURL Command**:



* 1. **Output**:  
     A black screen with red text

     Description automatically generated with low confidence