

Sending JSON to NodeJS to Kafka.

What we are trying to achieve ?

1. Send json from and browser/curl to nodejs.
2. nodejs will redirect json data to kafka.
3. Further processing is done on kafka.
4. We can then see the json arrive on kafka-console-consumer.sh script.

Step 1 : Create a script called json_nodejs_kafka.js with below script.

```
/*
    Getting some 'http' power
*/
var http=require('http');

/*
    Setting where we are expecting the request to arrive.
    http://localhost:8125/upload
*/
var request = {
    hostname: 'localhost',
    port: 8125,
    path: '/upload',
    method: 'GET'
};

/*
    Lets create a server to wait for request.
*/
http.createServer(function(request, response)
{
    /*
        Making sure we are waiting for a JSON.
    */
    response.writeHead(200, {"Content-Type": "application/json"});

    /*
        request.on waiting for data to arrive.
    */
    request.on('data', function (chunk)
    {

        /*
            CHUNK which we recive from the clients
            For out request we are assuming its going to be a JSON data.
        */
    }
    )
    )
    .listen(8125, 'localhost');
```

```

        We print it here on the console.
    */
    console.log(chunk.toString('utf8'))

    /*
        Using kafka-node - really nice library
        create a producer and connect to a Zookeeper to send the payloads.
    */
    var kafka = require('kafka-node'),
        Producer = kafka.Producer,
        client = new kafka.Client('kafka:2181'),
        producer = new Producer(client);

    /*
        Creating a payload, which takes below information
        'topic'      --> this is the topic we have created in kafka.
        'messages'   --> data which needs to be sent to kafka. (JSON in our case)
        'partition' --> which partition should we send the request to.
                        If there are multiple partition, then we optimize the code here,
                        so that we send request to different partitions.
    */

    payloads = [
    { topic: 'test', messages: chunk.toString('utf8'), partition: 0 },
    ];

    /*
        producer 'on' ready to send payload to kafka.
    */
    producer.on('ready', function(){
        producer.send(payloads, function(err, data){
            console.log(data)
        });
    });

    /*
        if we have some error.
    */
    producer.on('error', function(err){})

    });
    /*
        end of request
    */
    response.end();

    /*
        Listen on port 8125
    */
    }).listen(8125);

```

Step 2 : Start above script on the nodejs server.

```

[nodejs-admin@nodejs nodejs]$ vim json_nodejs_kafka.js
[nodejs-admin@nodejs nodejs]$ node json_nodejs_kafka.js

```

Step 3 : Execute curl command to send the JSON to nodejs.

```
[nodejs-admin@nodejs nodejs]$ curl -H "Content-Type: application/json" \  
-d '{"username":"xyz","password":"xyz"}' http://localhost:8125/upload
```

Step 4 : Output on nodejs console

```
[nodejs-admin@nodejs nodejs]$ node json_nodejs_kafka.js  
{"username":"xyz","password":"xyz"}  
{ test: { '0': 29 } }
```

{"username":"xyz","password":"xyz"} request from the curl command. { test: { '0': 29 } } response from the kafka cluster that, it has received the json.

Step5 : Output on the kafka consumer side.

```
[kafka-admin@kafka kafka_2.9.2-0.8.2.0]$ bin/kafka-console-consumer.sh \  
--zookeeper localhost:2181 --topic test --from-beginning  
{"username":"xyz","password":"xyz"}
```

{"username":"xyz","password":"xyz"} data received from nodejs server.