# Node.js - Formidable upload with XHR

Asked 8 years, 2 months ago Active 8 years, 2 months ago Viewed 9k times

9

11

I try to implement a simple XHR upload to Node.js (via Formidable). The problem is that if I set

```
xhr.setRequestHeader("Content-Type", "multipart/form-data");
```

node gives me error:

```
Error: bad content-type header, no multipart boundary
```

If I set boundary to just a random string nothing happens. The browser just hangs on POST and waits for server response.

The point is that if I use formidable with regular synchronous POST everything works fine.

Anyone tried to use Formidable with XHR upload?

node.js xmlhttprequest

asked Jul 30 '11 at 16:12

Pono

5,779 7 44 55

## 2 Answers

13

✓

I figured it out. I made a small bug on the client side.

Here is the **working expample of XHR upload with Formidable**

You **don't need** to set any boundaries or special headers.

### Client

```
var formData = new FormData();
var xhr = new XMLHttpRequest();

var onProgress = function(e) {
  if (e.lengthComputable) {
    var percentComplete = (e.loaded/e.total)*100;
  }
};

var onReady = function(e) {
 // ready state
};

var onError = function(err) {
  // something went wrong with upload
};
```

```
formData.append('files', file);
xhr.open('post', '/up', true);
xhr.addEventListener('error', onError, false);
xhr.addEventListener('progress', onProgress, false);
xhr.send(formData);
xhr.addEventListener('readystatechange', onReady, false);
```

### Server

```
app.post('/up', function(req, res) {
  var form = new formidable.IncomingForm();
  form.uploadDir = __dirname + '/tmp';
  form.encoding = 'binary';

  form.addListener('file', function(name, file) {
    // do something with uploaded file
  });

  form.addListener('end', function() {
    res.end();
  });

  form.parse(req, function(err, fields, files) {
    if (err) {
      console.log(err);
    }
  });
});
```

answered Jul 31 '11 at 14:42

Pono
**5,779** 7 44 55

I'm getting the error "bad content-type header, no content-type". Not really sure how to get round this? – wilsonpage Oct 6 '11 at 13:03

@pagewil Try setting enctype="multipart/form-data" on your <form> tag – Pono Oct 6 '11 at 20:39

Could you share what was actually the problem on the client side, which fixed it? – Alexander Jun 25 '14 at 7:11

Great! It helps. Thanks! Did you find out why can't specify the header type? – Brick Yang Nov 11 '15 at 8:20

2

I tried a couple of different things with Formidable and was never able to get it to accept a XHR upload. Here is what I ended up doing for a file upload using express.

```
app.post('/upload', function (req,res){
  if(req.xhr){
    var fSize = req.header('x-file-size'),
    fType = req.header('x-file-type'),
    basename = require('path').basename,
    fName = basename(req.header('x-file-name')),
    ws = fs.createWriteStream('./temp/'+fName);
    req.on('data', function(data) {
      ws.write(data);
    });
```

Hope this helps.

answered Jul 30 '11 at 19:19

scottmizo
**96** 2 4

1 I am able to get it to accept a XHR upload using Formidable. – Tareq Feb 16 '12 at 5:47