

## FE User Management - Bug #44438

### form validation with regexp fails with jsvalidator

2013-01-10 16:53 - Martin Muskulus

<b>Status:</b>	Resolved	<b>Start date:</b>	2013-01-10
<b>Priority:</b>	Should have	<b>Due date:</b>	
<b>Assignee:</b>	Bernhard Baumgartl, datamints GmbH	<b>% Done:</b>	100%
<b>Category:</b>		<b>Estimated time:</b>	0.00 hour
<b>Target version:</b>			

#### Description

The example code from the docs at [http://typo3.org/extension-manuals/datamints\\_feuser/0.8.3/view/1/4/#idm89295424](http://typo3.org/extension-manuals/datamints_feuser/0.8.3/view/1/4/#idm89295424) or #14047 to validate usergroup with a custom regexp validator fails when jsvalidator is used at the same time. I always get the message defined in plugin.tx\_datamintsfeuser\_pi1.\_LOCAL\_LANG.default.usergroup\_error\_valid.

Digging through JavaScript and PHP code I discovered that plugin.tx\_datamintsfeuser\_pi1.validate.usergroup.regexp is passed on to JavaScript's new RegExp() and to PHP's preg\_match() with the **same value**.

I expect "pattern" to be the string to search for only, i.e. it must not contain any delimiter. PHP's preg\_match() expects "/pattern/" as parameter (with delimiters), but JavaScript's RegExp() expects only "pattern" (without delimiters) and transforms it into "/pattern/" during processing regexp functions. This leads us to a wrong pattern "\/pattern\/" applied in validator.js line 228

```
if (!value[k].match(validate['regexp']))
```

[https://developer.mozilla.org/en-US/docs/JavaScript/Reference/Global\\_Objects/RegExp](https://developer.mozilla.org/en-US/docs/JavaScript/Reference/Global_Objects/RegExp) says:

Notice that the parameters to the literal format do not use quotation marks to indicate strings, while the parameters to the constructor function do use quotation marks. So the following expressions create the same regular expression:

```
/ab+c/i;  
new RegExp("ab+c", "i");
```

Imho, the correct place to solve the issue is in pi1/class.tx\_datamintsfeuser\_pi1.php around line line 2828

```
} else if ($key == 'regexp') {  
$configuration .= $this->extKey . '_config[' . $this->contentId . '][' . $fieldName . ']["validation"][' . $key . ']=new RegExp(" . $cleanedVal . ');';  
} else {  
$configuration .= $this->extKey . '_config[' . $this->contentId . '][' . $fieldName . ']["validation"][' . $key . ']=" . $cleanedVal . ');';  
}  
}
```

This current implementation doesn't pass on any pattern modifier as second argument of RegExp(). When using only the else branch all will work fine, if TypoScript contains

```
plugin.tx_datamintsfeuser_pi1.validate.usergroup.regexp = /pattern/
```

but not with

```
plugin.tx_datamintsfeuser_pi1.validate.usergroup.regexp = pattern
```

So there need to be some additional check implemented for people writing typoscript without delimiters otherwise PHP's preg\_match() will fail.

The attached patch is a temporary workaround in JavaScript to remain backwards compatibility and delete all / and \ from pattern in validator.js. I made a copy of validator.js and put plugin.tx\_datamintsfeuser\_pi1.jsvalidatorpath in place to not overwrite the workaround with the next extension update.

---

## History

---

### #1 - 2014-02-19 16:58 - Bernhard Baumgartl, datamints GmbH

- Status changed from New to Needs Feedback
- Assignee set to Bernhard Baumgartl, datamints GmbH
- % Done changed from 0 to 90

Fixed in r83680.

### #2 - 2014-03-07 16:23 - Martin Muskulus

Works for me. Thank you.

### #3 - 2014-07-18 18:04 - Bernhard Baumgartl, datamints GmbH

- Status changed from Needs Feedback to Resolved
- % Done changed from 90 to 100

## Files

---

validator.js.patch	353 Bytes	2013-01-10	Martin Muskulus
--------------------	-----------	------------	-----------------