QFQ - Support #12664

TinyMCE: report/remove malicous HTML/JS Code

09.06.2021 09:57 - Carsten Rose

Status:	New	**Start date:**	09.06.2021
Priority:	Normal	**Due date:**	
Assignee:	Carsten Rose	**% Done:**	0%
Category:		**Estimated time:**	0.00 hour
Target version:	22.3.0	**Spent time:**	0.00 hour
Discuss:			

Description

Damit TinyMCE sinnvoll genutzt wird, darf der Content nicht htmlspecialchar kodiert gespeichert werden.

Aber: wird der Code 1:1 gespeichert, koennte ein Angreifer Javascript injizieren und z.B. die PHP Session abgreifen (JS in einem HTML Attribut, oder einfach <script>...).

Es ist wichtig den Code zu checken und entweder zu bereinigen oder abzulehnen.

History

#1 - 09.06.2021 13:26 - Carsten Rose

Vorschlag:

- FE.typ=Editor/TinyMCE werden bei default 'bereinigt'
- FE.parameter.cleanHtml=1 (default) wenn Editor=TinyMCE.
- Der Parameter kann auf '0' gesetzt werden um das Cleanung abzuschalten.
- Es ist wichtig das ein 'Clean' gemacht wird und kein 'Reject' (also keine Exception werfen bei unlauterem Code). Grund:
  - Wenn aus einem Worddokument viel Content kopiert wird, soll dieser bereinigt gespeichert werden
  - Wuerde er zurueck gewiesen muesste, der User umstaendlich Stueck fuer Stueck versuchen den problematischen Teil zu finden - das wird niemand machen und stattdessen sagen 'unbrauchbares System'.

- Der Parameter cleanHtml=1 kann auch bei anderen FE.Elementen gesetzt werden. Dort ist der Default 0, und er muss aktiv gesetzt werden.
- Das setzen des Parameters macht nur Sinn bei FE.encode=none|single tick.

Anmerkung:

- Bekannte Probleme: https://html5sec.org/.
- Das entfernen / saeubern sollte nicht mit einem eigenen HTML Parser erfolgen (zu aufwendig, langsam)
- Genau unser Fall, aber keine Loesung: https://stackoverflow.com/questions/58759212/php-securing-a-users-full-html-file-against-xss
- Best Practice um einen DOM zu erstellen und zu bewerten: https://gist.github.com/lyquix-owner/9dd5eee80b8aaee2bd968e3a48641909
- Cleaning libs
  - HTML Purifier http://htmlpurifier.org/
  - htmLawed http://www.bioinformatics.org/phplabware/internal_utilities/htmLawed/index.php

- Check wie der DOM Parser auf kaputtes HTML reagiert. https://html5sec.org/#91

Order:

- 1) UTF8 Codes ersetzen (check ob der dom parser das bereits macht?)
- 2) lower case (check ob der dom parser das bereits macht?)
- 3) Blacklist Tags entfernen
- 4) Blacklist Attribute entferen
- 5) Alle Attribute die mit javascript:... beginnen (kann bei src und href benutzt werden), entfernen.

cleanHtml= 0|1
cleanHtmlAttribute = on.*,srcdoc,srcset
cleanHtmlAttributeMaxLength = 50 - Damit sollen ungewoehnlich lange Attribute entfernt werden.
cleanHtmlTag=script,button,source,iframe,comment,object

#2 - 07.12.2021 16:15 - Carsten Rose

- Target version changed from 22.2.0 to 22.3.0