blob: d88c0b94e31b5b4083493d36665b6967b43b0d7c (plain)

```
<?php
# Provide an URL with a valid filename that browsers will use (save as...)
#
# Copyright (C) 1999, 2000 The SourceForge Crew
# Copyright (C) 2001, 2002 Laurent Julliard, CodeX Team, Xerox
# Copyright (C) 2000-2006 Mathieu Roy <yeupou--gnu.org>
# Copyright (C) 2014, 2016, 2017 Assaf Gordon
# Copyright (C) 2001-2011, 2013, 2017 Sylvain Beucler
# Copyright (C) 2013, 2014, 2017-2026 Ineiev
#
# This file is part of Savane.
#
# Code written before 2008-03-30 (commit 8b757b2565ff) is distributed
# under the terms of the GNU General Public license version 3 or (at your
# option) any later version; further contributions are covered by
# the GNU Affero General Public license version 3 or (at your option)
# any later version.  The license notices for the AGPL and the GPL follow.
#
# Savane is free software: you can redistribute it and/or modify
# it under the terms of the GNU Affero General Public License as
# published by the Free Software Foundation, either version 3 of the
# License, or (at your option) any later version.
#
# Savane is distributed in the hope that it will be useful,
# but WITHOUT ANY WARRANTY; without even the implied warranty of
# MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
# GNU Affero General Public License for more details.
#
# You should have received a copy of the GNU Affero General Public License
# along with this program.  If not, see <https://www.gnu.org/licenses/>.
#
# Savane is free software: you can redistribute it and/or modify
# it under the terms of the GNU General Public License as
# published by the Free Software Foundation, either version 3 of the
# License, or (at your option) any later version.
#
# Savane is distributed in the hope that it will be useful,
# but WITHOUT ANY WARRANTY; without even the implied warranty of
# MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
# GNU General Public License for more details.
#
# You should have received a copy of the GNU General Public License
# along with this program.  If not, see <https://www.gnu.org/licenses/>.

$GLOBALS['skip_csp_headers'] = 1;

foreach (['init', 'http', 'form-check'] as $i)
  require_once ("include/$i.php");

function file_exit ($func, $param)
{
  unset ($GLOBALS['skip csp headers']);
```

```
  utils_set_csp_headers ();
  $func = "exit_$func";
  $func ($param);
}

function file_exit_error ($str)
{
  file_exit ('error', $str);
}

extract (sane_import ('request',
  [
    'preg' => [['file_id', '/^(\d+|test[.]png)$/']], 'digits' => 'file_uid'
  ]
));

if (!$file_id)
  file_exit ('missing_param', ['file_id']);

if ($file_id == 'test.png')
  {
    header ('Content-Type: image/png');
    $fname = $GLOBALS['sys_www_topdir'] . '/images/common/floating.png';
    header ('Content-Length: ' . stat ($fname)['size']);
    header ("Content-Disposition: attachment; filename=$file_id");
    readfile ($fname);
    exit (0);
  }

# Check privacy of the item this file is attached to and reject access by
# non-authorized users.

$result = db_execute (
  "SELECT item_id, artifact FROM trackers_file WHERE file_id = ?", [$file_id]
);

if (db_numrows ($result) > 0)
  {
    $item_id  = db_result ($result, 0, 'item_id');
    $artifact = db_result ($result, 0, 'artifact');
  }
else
  # TRANSLATORS: the argument is file id (a number).
  file_exit_error (sprintf (_("File #%s not found"), $file_id));

$in = [0 => $artifact];
$out = [];

if ($sane_sanitizers['artifact'] ($in, $out, 0, null))
  {
    # TRANSLATORS: the argument is artifact name ('bugs', 'task' etc.)
    $str = sprintf (_('Invalid artifact %s'), "<em>$artifact</em>");
    unset ($artifact);
    file_exit_error ($str);
  }

function assert_file_access ($item_fields, $file_uid)
{
  if ($item_fields['privacy'] != '2')
    return;
  if (user_can_be_super_user ($file_uid))
    # We are in the file domain and have no access to cookies, so we can't tell
    # if the user has become a superuser; therefore, we let site admins access
    # any files in any case.
    return;
```

```
118   $group_id = $item_fields['group_id'];
119   if (!member_check_private ($file_uid, $group_id))
120     file_exit_error (
121       _("Non-authorized access to file attached to private item")
122     );
123   form_check_id ();
124 }
125 
126 $item_fields = utils_find_item (
127   $artifact, $item_id, ['privacy'], 'file_exit_error'
128 );
129 assert_file_access ($item_fields, $file_uid);
130 
131 $result = db_execute ("
132   SELECT description, filename, filesize, filetype, date
133   FROM trackers_file WHERE file_id = ? LIMIT 1", [$file_id]
134 );
135 
136 if (!db_numrows ($result))
137   file_exit_error (
138     sprintf (_("Couldn't find attached file #%s."), $file_id)
139   );
140 
141 $row = db_fetch_array ($result);
142 
143 if ($row['filesize'] < 0)
144   file_exit_error (
145     sprintf (_("Attached file #%s was lost."), $file_id) . " "
146     . sprintf (
147         _("File attributes: name '%s', size %s, type '%s', date %s."),
148         $row['filename'], $row['filesize'], $row['filetype'],
149         utils_format_date ($row['date'])
150       )
151   );
152 $mtime = $row['date'];
153 http_exit_if_not_modified ($mtime);
154 header ('Last-Modified: ' . date ('r', $mtime));
155 
156 # Check if the filename in database matches the one in the URL.
157 # We do not want to allow broken URL that may make a user download
158 # a file with a given name like "myimage.png" when actually downloading
159 # something completely different like "mystupidvirus.scr".
160 if ($row['filename'] != basename (rawurldecode ($_SERVER['PHP_SELF'])))
161   file_exit_error (
162     _("The filename in the URL does not match the filename "
163       . "registered in the database")
164   );
165 
166 $path = "$sys_trackers_attachments_dir/$file_id";
167 if (!is_readable ($path))
168   file_exit_error (_("No access to the file."));
169 
170 $row['agpl'] = trim (git_agpl_notice ('This file is served with Savane.'));
171 
172 # Serve the file with respective attributes.
173 $headers = [
174   'filetype' => 'Content-Type: ', 'filesize' => 'Content-Length: ',
175   'filename' => 'Content-Disposition: attachment; filename=',
176   'description' => 'Content-Description: ', 'agpl' => 'Source-Code-Offer: '
177 ];
178 foreach ($headers as $field => $h)
179   {
180     if (empty ($row[$field]))
181       continue;
182     $val = str_replace (["\n", "\r"], ' ', $row[$field]);
```

```
183     header ("$h$val");
184   }
185 readfile ($path);
186 exit (0);
187 ?>
```