

Extensions must meet the following rules

by [votdev](#) » Tue Jul 14, 2009 7:18 am

Hi all,

For FreeNAS 0.7 all extensions must meet the following implementation rules (i will check how to block them if they do not do that):

1. include auth.inc in the first line of your code

Code: [Select all](#)

```
OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE
POSSIBILITY OF SUCH DAMAGE.
*/
require("auth.inc");      <=====
require("guiconfig.inc");

$uuid = $_GET['uuid'];
if (isset($_POST['uuid']))
    $uuid = $_POST['uuid'];
```

2. Include formend.inc before the </form> closure

Code: [Select all](#)

```
<?php $options = get_fstype_list();?>
    <?php html_combobox("fstype", gettext("Preformatted file system"), $pconfig['fstype'], $options, gettext("This allows
you to set the file system for preformatted hard disks containing data.") . " " . sprintf(gettext("Leave '%s' for unformatted disks
and format them using <a href=%s>format</a> menu."), "Unformatted", "disks_init.php"), false);?>
    </table>
    <div id="submit">
        <input name="Submit" type="submit" class="formbtn" value="<?=(isset($uuid) && (FALSE !== $cnid)) ? gettext("Save") :
gettext("Add")?>" onClick="enable_change(true)">
        <input name="Cancel" type="submit" class="formbtn" value="<?=gettext("Cancel");?>">
        <input name="uuid" type="hidden" value="<?=$pconfig['uuid'];?>">
    </div>
    <?php include("formend.inc");?>      <=====
    </form>
</td>
</tr>
</table>
<?php if (isset($uuid) && (FALSE !== $cnid)):?>
<script language="JavaScript">
<!-- Disable controls that should not be modified anymore in edit mode. -->
enable_change(false);
</script>
<?php endif;?>
<?php include("fend.inc");?>
```

Check out the following example:

Code: [Select all](#)

```
#!/usr/local/bin/php
<?php
/*
services_bittorrent.php
Modified for XHTML by Daisuke Aoyama (aoyama@peach.ne.jp)
Copyright (C) 2010 Daisuke Aoyama <aoyama@peach.ne.jp>.
All rights reserved.
```

Copyright (c) 2006-2010 Volker Theile (votdev@gmx.de)
All rights reserved.

part of FreeNAS (<http://www.freenas.org>)
Copyright (C) 2005-2010 Olivier Cochard <olivier@freenas.org>.
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice,
this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES,
INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY
AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE
AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY,
OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE
POSSIBILITY OF SUCH DAMAGE.

```
*/
require("auth.inc");
require("guiconfig.inc");
require("services.inc");

$pgtitle = array(gettext("Services"), gettext("BitTorrent"));

if (!is_array($config['bittorrent']))
    $config['bittorrent'] = array();

$pconfig['enable'] = isset($config['bittorrent']['enable']);
$pconfig['port'] = $config['bittorrent']['port'];
$pconfig['downloadaddir'] = $config['bittorrent']['downloadaddir'];
$pconfig['configdir'] = $config['bittorrent']['configdir'];
$pconfig['username'] = $config['bittorrent']['username'];
$pconfig['password'] = $config['bittorrent']['password'];
$pconfig['authrequired'] = isset($config['bittorrent']['authrequired']);
$pconfig['peerport'] = $config['bittorrent']['peerport'];
$pconfig['portforwarding'] = isset($config['bittorrent']['portforwarding']);
$pconfig['uplimit'] = $config['bittorrent']['uplimit'];
$pconfig['downlimit'] = $config['bittorrent']['downlimit'];
$pconfig['pex'] = isset($config['bittorrent']['pex']);
$pconfig['dht'] = isset($config['bittorrent']['dht']);
$pconfig['encryption'] = $config['bittorrent']['encryption'];
$pconfig['watchdir'] = $config['bittorrent']['watchdir'];
$pconfig['incompletedir'] = $config['bittorrent']['incompletedir'];
$pconfig['umask'] = $config['bittorrent']['umask'];
$pconfig['extraoptions'] = $config['bittorrent']['extraoptions'];

// Set default values.
if (!$pconfig['port']) $pconfig['port'] = "9091";

if ($_POST) {
    unset($input_errors);
    $pconfig = $_POST;

    // Input validation.
    if ($_POST['enable']) {
        $reqdfields = explode(" ", "port downloadaddir peerport");
        $reqdfieldsn = array(gettext("Port"), gettext("Download directory"), gettext("Peer port"));
        $reqdfieldst = explode(" ", "port string port");
```

```

if ($_POST['authrequired']) {
// !!! Note !!! It seems TransmissionBT does not support special characters,
// so use 'alias' instead of 'password' check.
$reqdfields = array_merge($reqdfields, explode(" ", "username password"));
$reqdfieldsn = array_merge($reqdfieldsn, array(gettext("Username"), gettext("Password")));
$reqdfieldst = array_merge($reqdfieldst, explode(" ", "alias alias"));
}

do_input_validation($_POST, $reqdfields, $reqdfieldsn, &$input_errors);

// Add additional type checks
if (isset($_POST['umask'])) {
$reqdfields = array_merge($reqdfields, explode(" ", "umask"));
$reqdfieldsn = array_merge($reqdfieldsn, array(gettext("User mask")));
$reqdfieldst = array_merge($reqdfieldst, explode(" ", "filemode"));
}

do_input_validation_type($_POST, $reqdfields, $reqdfieldsn, $reqdfieldst, &$input_errors);

// Check if port is already used.
if (services_is_port_used($_POST['port'], "bittorrent")) {
$input_errors[] = sprintf(gettext("Port %ld is already used by another service."), $_POST['port']);
}

// Check port range.
if ($_POST['port'] && ((1024 > $_POST['port']) || (65535 < $_POST['port']))) {
$input_errors[] = sprintf(gettext("The attribute '%s' must be in the range from %d to %d."), gettext("Port"), 1024,
65535);
}
}

if (!$input_errors) {
$config['bittorrent']['enable'] = $_POST['enable'] ? true : false;
$config['bittorrent']['port'] = $_POST['port'];
$config['bittorrent']['downloaddir'] = $_POST['downloaddir'];
$config['bittorrent']['configdir'] = $_POST['configdir'];
$config['bittorrent']['username'] = $_POST['username'];
$config['bittorrent']['password'] = $_POST['password'];
$config['bittorrent']['authrequired'] = $_POST['authrequired'] ? true : false;
$config['bittorrent']['peerport'] = $_POST['peerport'];
$config['bittorrent']['portforwarding'] = $_POST['portforwarding'] ? true : false;
$config['bittorrent']['uplimit'] = $_POST['uplimit'];
$config['bittorrent']['downlimit'] = $_POST['downlimit'];
$config['bittorrent']['pex'] = $_POST['pex'] ? true : false;
$config['bittorrent']['dht'] = $_POST['dht'] ? true : false;
$config['bittorrent']['encryption'] = $_POST['encryption'];
$config['bittorrent']['watchdir'] = $_POST['watchdir'];
$config['bittorrent']['incompletedir'] = $_POST['incompletedir'];
$config['bittorrent']['umask'] = $_POST['umask'];
$config['bittorrent']['extraoptions'] = $_POST['extraoptions'];

write_config();

$retval = ;
if (!file_exists($d_sysrebootreqd_path)) {
config_lock();
$retval |= rc_update_service("transmission");
$retval |= rc_update_service("mdnsresponder");
config_unlock();
}

$savemsg = get_std_save_message($retval);
}
}
?>
<?php include("fbegin.inc");?>
<script type="text/javascript">

```

```

<!--
function enable_change(enable_change) {
var endis = !(document.iform.enable.checked || enable_change);
document.iform.port.disabled = endis;
document.iform.download.dir.disabled = endis;
document.iform.download.dir.browser.btn.disabled = endis;
document.iform.config.dir.disabled = endis;
document.iform.config.dir.browser.btn.disabled = endis;
document.iform.authrequired.disabled = endis;
document.iform.username.disabled = endis;
document.iform.password.disabled = endis;
document.iform.peerport.disabled = endis;
document.iform.port.forwarding.disabled = endis;
document.iform.uplimit.disabled = endis;
document.iform.downlimit.disabled = endis;
document.iform.pex.disabled = endis;
document.iform.dht.disabled = endis;
document.iform.encryption.disabled = endis;
document.iform.watch.dir.disabled = endis;
document.iform.watch.dir.browser.btn.disabled = endis;
document.iform.incomplete.dir.disabled = endis;
document.iform.incomplete.dir.browser.btn.disabled = endis;
document.iform.umask.disabled = endis;
document.iform.extraoptions.disabled = endis;
}

function authrequired_change() {
switch (document.iform.authrequired.checked) {
case true:
showElementById('username_tr','show');
showElementById('password_tr','show');
break;

case false:
showElementById('username_tr','hide');
showElementById('password_tr','hide');
break;
}
}
//-->
</script>
<form action="services_bittorrent.php" method="post" name="iform" id="iform">
<table width="100%" border="0" cellpadding="0" cellspacing="0">
<tr>
<td class="tabcont">
<?php if ($input_errors) print_input_errors($input_errors);?>
<?php if ($savemsg) print_info_box($savemsg);?>
<table width="100%" border="0" cellpadding="6" cellspacing="0">
<?php html_titleline_checkbox("enable", gettext("BitTorrent"), $pconfig['enable'] ? true : false, gettext("Enable"),
"enable_change(false"));?>
<?php html_inputbox("peerport", gettext("Peer port"), $pconfig['peerport'], sprintf(gettext("Port to listen for
incoming peer connections. Default port is %d."), 51413), true, );?>
<?php html_filechooser("download.dir", gettext("Download directory"), $pconfig['download.dir'], gettext("Where to
save downloaded data."), $g['media_path'], true, 60);?>
<?php html_filechooser("config.dir", gettext("Configuration directory"), $pconfig['config.dir'], gettext("Alternative
configuration directory (usually empty)."), $g['media_path'], false, 60);?>
<?php html_checkbox("portforwarding", gettext("Port forwarding"), $pconfig['portforwarding'] ? true : false,
gettext("Enable port forwarding via NAT-PMP or UPnP."), "", false);?>
<?php html_checkbox("pex", gettext("Peer exchange"), $pconfig['pex'] ? true : false, gettext("Enable peer exchange
(PEX)."), "", false);?>
<?php html_checkbox("dht", gettext("Distributed hash table"), $pconfig['dht'] ? true : false, gettext("Enable
distributed hash table."), "", false);?>
<?php html_combobox("encryption", gettext("Encryption"), $pconfig['encryption'], array("0" => gettext("Tolerated"),
"1" => gettext("Preferred"), "2" => gettext("Required")), gettext("The peer connection encryption mode."), false);?>
<?php html_inputbox("uplimit", gettext("Upload bandwidth"), $pconfig['uplimit'], gettext("The maximum upload
bandwidth in KB/s. An empty field means infinity."), false, );?>
<?php html_inputbox("downlimit", gettext("Download bandwidth"), $pconfig['downlimit'], gettext("The maximum
download bandwidth in KiB/s. An empty field means infinity."), false, );?>

```

```

<?php html_filechooser("watchdir", gettext("Watch directory"), $pconfig['watchdir'], gettext("Directory to watch for
new .torrent files."), $g['media_path'], false, 60);?>
<?php html_filechooser("incompletedir", gettext("Incomplete directory"), $pconfig['incompletedir'],
gettext("Directory to incomplete files. An empty field means disable."), $g['media_path'], false, 60);?>
<?php html_inputbox("umask", gettext("User mask"), $pconfig['umask'], sprintf(gettext("Use this option to override
the default permission modes for newly created files (%s by default)."), "0002"), false, );?>
<?php html_inputbox("extraoptions", gettext("Extra options"), $pconfig['extraoptions'], gettext("Extra options
(usually empty).") . " " . sprintf(gettext("Please check the <a href='%s' target='_blank'>documentation</a>."),
"http://trac.transmissionbt.com/wiki/man/transmission-remote"), false, 40);?>
<?php html_separator();?>
<?php html_titleline(gettext("Administrative WebGUI"));?>
<?php html_inputbox("port", gettext("Port"), $pconfig['port'], sprintf(gettext("Port to listen on. Default port is %d."),
9091), true, );?>
<?php html_checkbox("authrequired", gettext("Authentication"), $pconfig['authrequired'] ? true : false,
gettext("Require authentication."), "", false, "authrequired_change());?>
<?php html_inputbox("username", gettext("Username"), $pconfig['username'], "", true, 20);?>
<?php html_passwordbox("password", gettext("Password"), $pconfig['password'], gettext("Password for the
administrative pages."), true, 20);?>
<?php
$if = get_ifname($config['interfaces']['lan']['if']);
$ipaddr = get_ipaddr($if);
$url = htmlspecialchars("http://{ $ipaddr } : { $pconfig['port'] }");
$text = "<a href='{ $url }' target='_blank'>{ $url } </a>";
?>
<?php html_text("url", gettext("URL"), $text);?>
</table>
<div id="submit">
<input name="Submit" type="submit" class="formbtn" value="<?=gettext("Save and Restart");?>"
onclick="enable_change(true)" />
</div>
</td>
</tr>
</table>
<?php include("formend.inc");?>
</form>
<script type="text/javascript">
<!--
enable_change(false);
authrequired_change();
/-->
</script>
<?php include("fend.inc");?>

```

Regards
Volker