

## Patching Joomla 1.0.x to support AOL users

Did you received complaints from your AOL users that they have problems using your Joomla powered site?  
Did you read in Joomla forums that other users and administrators have the same problem, namely that the last few 1.0.x releases of Joomla do not properly support AOL users?

I did, and searched the web for some solution. This way I had came across the Eric's blog, where I found the following hack, which worked for me.

Here's what Eric has to say about the problem:

This issue stems from a misunderstanding of the AOL proxy layer. In short, AOL users can appear to use any of the "AOL Client/Proxy IP Addresses" listed here. Moreover, an individual user's IP can flip between a Client IP and a Proxy IP from one request to the next. Therefore, it is simply impossible to use an AOL client's IP address as a (primary) source of identification.

To address this, the current Joomla 1.0.x codebase supports a "Proxied User" session id hash [aka: Security Level 2] that uses a client's subnet instead of the full IP. This works for most proxied users, but is not sufficient for AOL users as they are not necessarily bound to a particular subnet.

I've hacked together a quick patch to properly handle AOL users. To use it, you'll need to modify your /includes/joomla.php file. I've modified one existing method and added a method to the file. Here's the code:

```
/*
 * Static Function used to generate the Session Cookie Value
 * Added as of 1.0.8
 * Deprecated 1.1
 */
function sessionCookieValue( $id=null ) {
global $mainframe;
$type = $mainframe->getCfg( 'session_type' );
$browser = @$_SERVER['HTTP_USER_AGENT'];
switch ( $type ) {
case 2:
// 1.0.0 to 1.0.7 Compatibility
// lowest level security
$value = md5( $id . @$_SERVER['REMOTE_ADDR'] );
break;
case 1:
// slightly reduced security-3rd level IP authentication for those behind IP Proxy
/**
 * AOL users are not supported by the default case handling because their client
 * IPs can change upon each request. Thus, we can NOT use their IP
 * in our mosHash. I understand the security implications (bad) and agree
 * that this AOL behavior is harmful, but we need to support AOL users and thus
 * this logic is necessary. ets_tempest
 */

```

```

if ( mosMainFrame::isAOLuser($browser,$_SERVER['REMOTE_ADDR']) ) {
$value = mosHash( $id . $browser );
}else{
$remote_addr = explode('.',$_SERVER['REMOTE_ADDR']);
$ip = $remote_addr[0] . '.' . $remote_addr[1] . '.' . $remote_addr[2];
$value = mosHash( $id . $ip . $browser );
}
break;
default:
// Highest security level - new default for 1.0.8 and beyond
$ip = $_SERVER['REMOTE_ADDR'];
$value = mosHash( $id . $ip . $browser );
break;
}
return $value;
}
/**
* The list below holds an array of the possible subnets for AOL users as
* declared here: http://webmaster.info.aol.com/proxyinfo.html
*
* This list can change and should be regularly updated. Thus, it would best be
* externalized.
*/
var $aolSubnetList = array('64.12.96','149.174.160','152.163.240','152.163.248','152.163.252',
'152.163.96','152.163.100','195.93.32','195.93.48','195.93.64','195.93.96','195.93.16','198.81',
'198.81.16','198.81.8','202.67.64','205.188.192','205.188.208','205.188.112','205.188.146',
'207.200.112','172.128','172.192','172.208','202.67.66','172.200','172.202','172.212','172.216',
'202.67.68','202.67.72','202.67.80','202.67.96');
/**
* Use agent headers and IP address comparison to detect AOL users
*/
function isAOLuser($browser,$clientIp){
if( strpos($browser, 'AOL') !== false ){
return true;
}
foreach($this->aolSubnetList as $aolSubnet){
if( strpos($clientIp,$aolSubnet) === 0 ){
return true;
}
}
return false;
}

```

That's it, folks! Worked for me, the complaints are a past issue. But, as allways: use at your own risk! No guarantees that will work in all environments and all setups.

But before don't forget to save everything, cross your fingers and put your pants inversely to be sure that everything will working as expected! Erik submitted it to the core team, but who know, if will be included or not in the next release.