Communications of the IIMA

Volume 11 | Issue 1 Article 5

2011

Discovering a Joomla Exploit for Possible Malware: Social Engineering and a PHP BASE64 GIF Exploit

M.S. Terrell Rohm
The Active Network, Inc.

C.E. Tapie Rohm Jr.
California State University San Bernardino

Haakon Brown
California State University San Bernardino

Discovering a Joomla Exploit for Possible Malware: Social Engineering and a PHP BASE64 GIF Exploit

M. S. Terrell Rohm
The Active Network, Inc.
USA
terrell.rohm@gmail.com

C. E. Tapie Rohm Jr.
California State University San Bernardino
USA
trohm@csusb.edu

Haakon Brown
California State University San Bernardino
USA
hbrown@csusb.edu

ABSTRACT

This article discusses the importance of Joomla as a Content Management System that is used by 2.7% of the web and how a possible new malware exploit has been discovered. The PHP BASE64 malware exploit is a well-documented exploit of PHP but the implementation of this exploit as it relates to Joomla is a very ingenious method not previously used before, as far as the authors were able to discover. In this particularly case, PHP code is embedded in a GIF file to produce a very sophisticated and unique malware exploit to Joomla.

INTRODUCTION

Joomla (2011) is a free Open Source software (Wheeler, 2007) solution that "is an award-winning content management system (CMS)." Joomla is used worldwide from personal to corporate environments such as:

- Corporate Web sites or portals,
- Corporate intranets and extranets,
- Online magazines, newspapers, and publications,
- E-commerce and online reservations,
- Government applications,
- Small business Web sites,
- Non-profit and organizational Web sites,
- Community-based portals,
- School and church Web sites, and
- Personal or family homepages.

According to the Joomla website, Joomla has been adopted by approximately 2.7% of the entire web as their CMS (Joomla, 2011) while Buildwith (2011) lists 1,408,800 web servers using the Joomla product. This is not the largest share of the market as WordPress (2011) has this distinction with 4,063,871 websites using WordPress or 4.57% share of the market. However, it is significant because many organizations and individuals rely on Joomla. Since Joomla is used as a mission critical tool, it is extremely important to keep it protected from malware exploits.

A malware exploit, "is a piece of software that attacks a particular security vulnerability (Global Oneness, 2011)." Joomla has been exploited since its development. Starting with the Mambo exploit listed as 2006-04-19 at the Joomla Exploit website, there are approximately 1,000 known or published exploits and vulnerabilities (JoomlaExploits, 2011).

The PHP BASE64 exploits have been documented by Winders (2010) and others (Jelsoft, 2010; Oscommerce.com, 2010). The PHP BASE64 GIF exploit is the result of embedding executable PHP code within the binary data of a GIF file. The desired malware PHP code is converted to binary through the PHP function base64_encode. The newly converted PHP code can then be copied into the binary data of a GIF file. Thus when the GIF file is loaded from the server, with the base64_decode function being called on the malicious code, the PHP code is encoded back to PHP and executed.

A Google search for downloadable Joomla templates returns over 100,000,000 results. As Joomla administrators look to utilize these downloadable Joomla templates they do so fully aware that these templates may contain malware or other malicious code. Joomla administrators will need to carefully review the source code of the Joomla templates for malware. However, review of Joomla template source code requires working knowledge of PHP and all possible PHP exploits. If Joomla administrators are not trained in PHP they may fall victim to Joomla templates with PHP exploits. Even those Joomla administrators that are well trained in PHP may not find the malicious code due to the malware author's effective use of hiding the exploits in the source as was the case in the "Spa Complex" Joomla available for download on hundreds of sites. http://www.mightyjoomla.com/free-joomla-template/spa-complex-free-joomla-template.

The malware author used a PHP BASE64 GIF exploit embedded within the Joomla template source code and images files to add a non-removable navigation link at the top of the template that links back to their sponsoring website. While this exploit is hardly malicious is could have been used for more malicious intent such as gathering website tracking or gathering user data entered on the website. The malware author could have hid his few lines of malicious code deep within Joomla's template hoping the code would either not be discovered or relying upon the Joomla's administrator's lack of PHP knowledge. However, the author's ingenious method for hiding the code was to obfuscate and break up the malicious source code over a number of lines thus making it incredibly difficult to track down and remove. Below is a complete breakdown of the source code with the malicious code highlighted. Notice how the code is not only broken down over multiple lines but obfuscated by hiding the exploited GIF's filename through numerous PHP variables.

HOW IT WORKS

According to Joomla (2010), Joomla code should start with the following PHP code at the top of the file:

defined('_JEXEC') or die('Restricted access');

This code is necessary because "This statement checks to see if the file is being called from within a Joomla session. This protects your site by making it more difficult for a cracker/hacker to damage your site." However, there is additional code, marked in bold, adding additional functionality to our free Joomla template.

Line 2 of the template index.php file says:

*defined('_JEXEC')**.(($this->template)?$JPan = array('zrah'.'_pby'):'')** or die('Restricted access');*

According to the user "kencmd" on http://forum.joomla.org/viewtopic.php?p=1673083 the bolded code is doing the following:

"**$JPan = array('zrah'.'_pby'):'')** =

Take zrah'.'_pby and remove the middle '.' which is there just to breakup the file name.

And you have 'zrah_pby'

ROT13 decode that and you have '**menu_col**'

Replace $JPan in this line and again remove the '.' connectors used just to break-up the strings

So str_rot13**($JPan[0]**.'.t'.'vs') = **menu_col.gif**

Now you have the file name."

Thus the added code to the beginning was nothing malicious as it seems it was a very obfuscated way to hide a PHP attribute to reference "menu_col.gif." Upon searching the images directory of the free Joomla template we find that menu_col.gif is actually an image. Nothing appears to be malicious but should definitely be considered suspicious.

Further down our suspicions are raised as we see a reference to $JPan[0].

*!@include(JPATH_BASE.DS.'templates'.DS.$mainframe->getTemplate().DS.str_rot13('vzntrf').DS.str_rot13**($JPan[0]**.'.t'.'vs'))) : >*

Breaking this down, we see that the code is creating a path from the Joomla template root:

JPATH_BASE.DS.'templates'.DS.$mainframe->getTemplate().DS
JPATH_BASE = Joomla application directory, "~/public_html"
.DS = Directory Separator (/)
.'templates' = A standard subdirectory for joomla templates
.DS = Directory Separator (/)
.$mainframe->getTemplate() = Returns the current name of the template, "themze_j15_40"
.DS = Directory Separator (/)

Thus the code has created the path of "~/public_html/templates/themza_j15_40/." The remaining piece of code is deciphered by user "kencmd.

"str_rot13('vzntrf') = 'images'
ROT13 decode that and you have another part of the path.

So all this is used just to hide making the path:

<joomla_root>/templates/template_name/images/menu_col.gif"

A closer look at this .gif file reveals you cannot open it in an internet browser nor a picture viewing program such as Picassa or MS Paint. Opening the file in MS Notepad reveals PHP code.

<?php */*GIF89a_ _ чЋ S›/цр Г›;рЯЌo_X±QS™+мЬSИ¦OпЫ|mаЋ3цк_нЮґKkйФ#_ це@ЬИ§нЭ_lX_oS*

*ff3ыл3мШууи_эн8__*ьк>аЖrTi„П®аИ\LЛ§Fяр_TI[§§#ЬВ%Щ¶ЬЩГ'!%(С¶!Ц№sъа оК·—ЦЄІ›Ц_пж—аЙ•яп!Лр_XR:щг.ШЕѓсФЈЖ§_LJ5i™DъмЇцжН__3аФs 3юи_dN_•,_ЭsdSЋLсгN3§_яп_Н«0I'_оаJ'v_з?ях_кЬsФs,ЯВЋлЩ_ио3ях_П¬Rчжћяр ОВ¬%++5i¤_йЧ‹йХvыф_Х№_УIIЭРS·— 4сгњЙ©^Ф·ВбЛ¤№™6Вў'БЈ3цр_Ps ✔nЯ'ЫО®nЮншф яя яяя !щ__ Ћ , _ _ _н _ _и_FЛЬ9-_*_ь,Тё_А—_*

ф/**function tdo(){echo base64_decode**('RGVzaWduZWQgYnk6IDxhIGhyZWY9Imh0dHA6Ly93d3cudGhlbXphLmNvbS8iIHRpdGxlPSIiIHRhcmdldD0iX2JsYW5rIj5mcmVlIEpvb21sYSAxLjUgdGhlbWU8L2E+ICA8YSBocmVmPSJodHRwOi8vd3d3LnJlc2VsbGVyc3BhbmVsLmNvbS9kZWRpY2F0ZWQtc2VydmVycy9kZWRpY2F0ZWQtbW9udGhseS10cmFmZmljLmh0bWwiIHRpdGxlPSIiIHRhcmdldD0iX2JsYW5rIj48L2E+Jm5ic3A7');}$GLOBALS['arr'] = array('name'=>'joomla web hosting','link_title'=>'','link'=>'http://www.ntchosting.com/','id'=>500,'menutype'=>'mainmenu','alias'=>'jj','type'=>'url','published'=>1,'parent'=>1,'componentid'=>0,'sublevel'=>1,'ordering'=>1,'checked_out'=>'0','pollid'=>'0','browserNav'=>1,'access'=>'0','utaccess'=>'0','params'=>array('menu_image'=>'-1'),'lft'=>0,'rtf'=>0,'home'=>0,'component'=>'','tree'=>array('0'=>'1','1'=>500),'check

ed_out_time'=>'0000-00-00
00:00:00','route'=>'home/jj','query'=>array('tst'=>''));{$a=strlen('');}/'' ✔HpM_:_*
Ш_„ЃJJ(_p_ЪaE ✔_&Yф€AE,‹,uЂђ ✔8ИБђ _Вш°_bH–

*sMш2aJ_(L*t__‡,B7d___увNњ"g>___0Ђ„ѓ*

9__0bdb`™__Ъ_Рр!Г‡С ∂6вЃИ‘"_Ђ__
Ђе…_<Z_5љЬЁР'@y_y0 @__)G_Рћьгк_'_Р,_y¤_

1J'3ъСаД„Е

j ✔АБf.J А—pѓ ‰Ђг_ (Wћ 9!_Ћ__ ;/?>*

Embedded within the .gif is a PHP function *t*do that *echo* or prints the results returned from the function *base64_decode*. The data being passed into the function is *base64_encoded* to further obfuscate and complicate our ability to determine its intent. Executing this function returns:

*Designed by: Joomla Templates, web hosting. *

This is an example of the well documented Gif images security exploit in PHP (Lemos, 2007). Luckily the suspicious yet evidently non-malicious code simply hard codes a link menu item to the top navigation of the website that could not be removed through the Joomla administrator. This is a very obfuscated and sneaky way to force Joomla users who are not familiar with PHP to keep a link in the navigation.

The real scary exploit identified here is not the GIF images security exploit which has been well documented along with documented defenses. The exploit incurs from a social engineering aspect of non-PHP/non-coders utilizing tools such as Joomla, Drupal and WordPress to create websites from "free" templates. These users can become victims as they download free templates for their Joomla, Drupal or WordPress sites and trust them to be non-malicious. Potentially embedded within these templates are exploits such as the Gif images security exploit, cross-sight scripting, traffic sniffing, or website analytic data.

Simple Diagram of the Malicious Code

The following three figures are visual diagrams of the exploit and how the code work together, see Figures 1, 2 & 3. The actual PHP code is listed in Appendix A.

Figure 1: Diagram of the Joomla PHP BASE64 GIF Configuration Exploit.

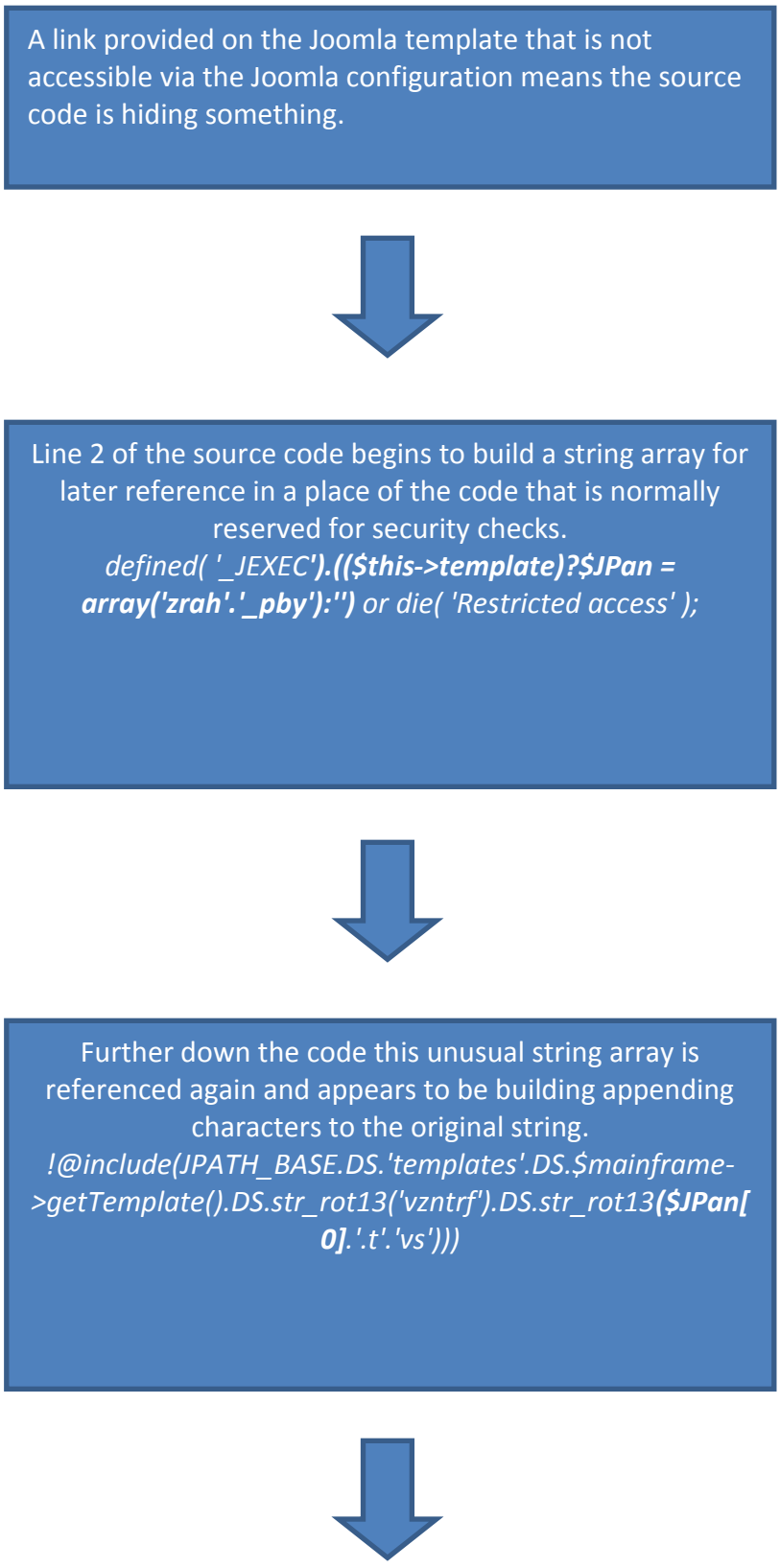


Figure 2: PHP BASE64 GIF Exploit Displayed.

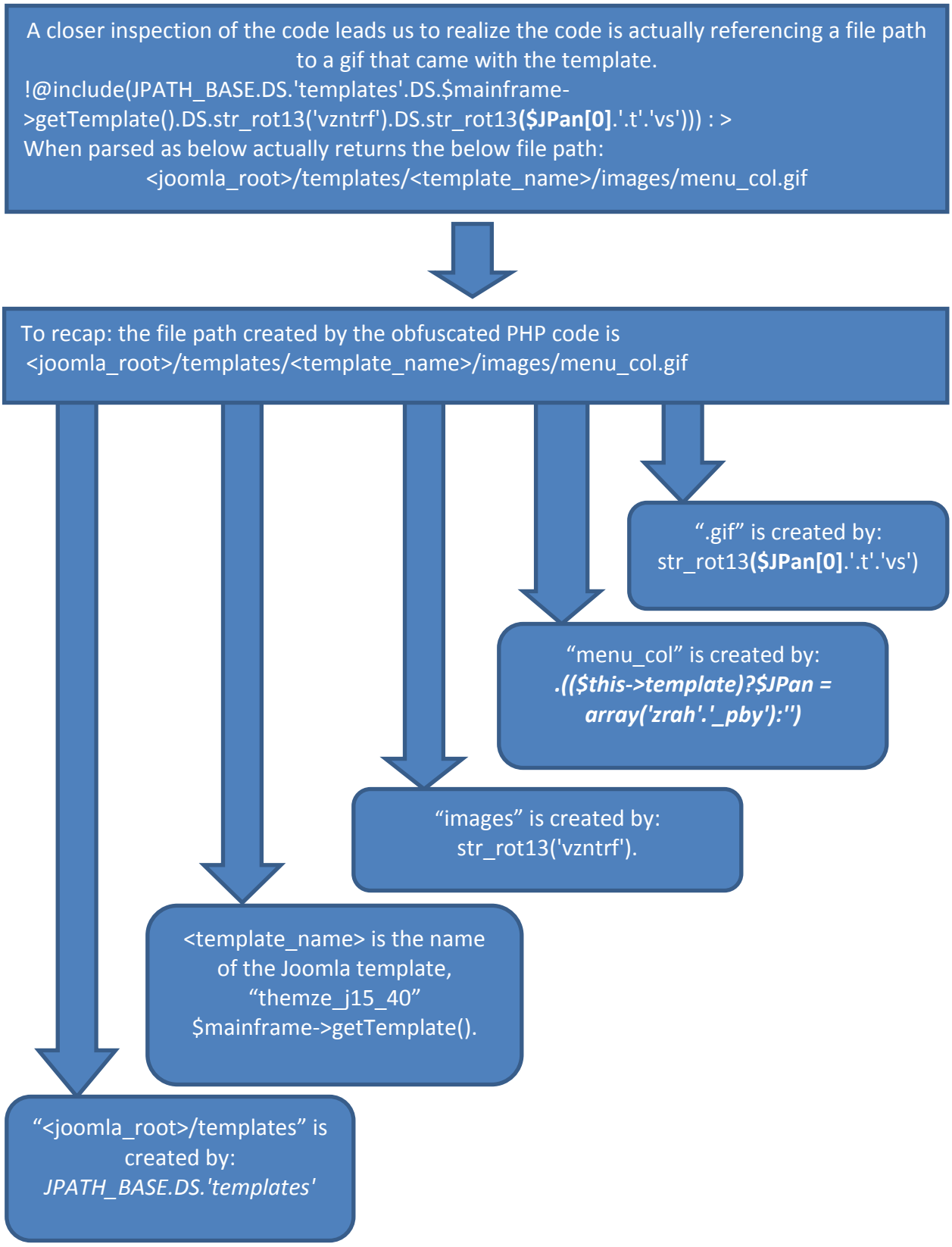
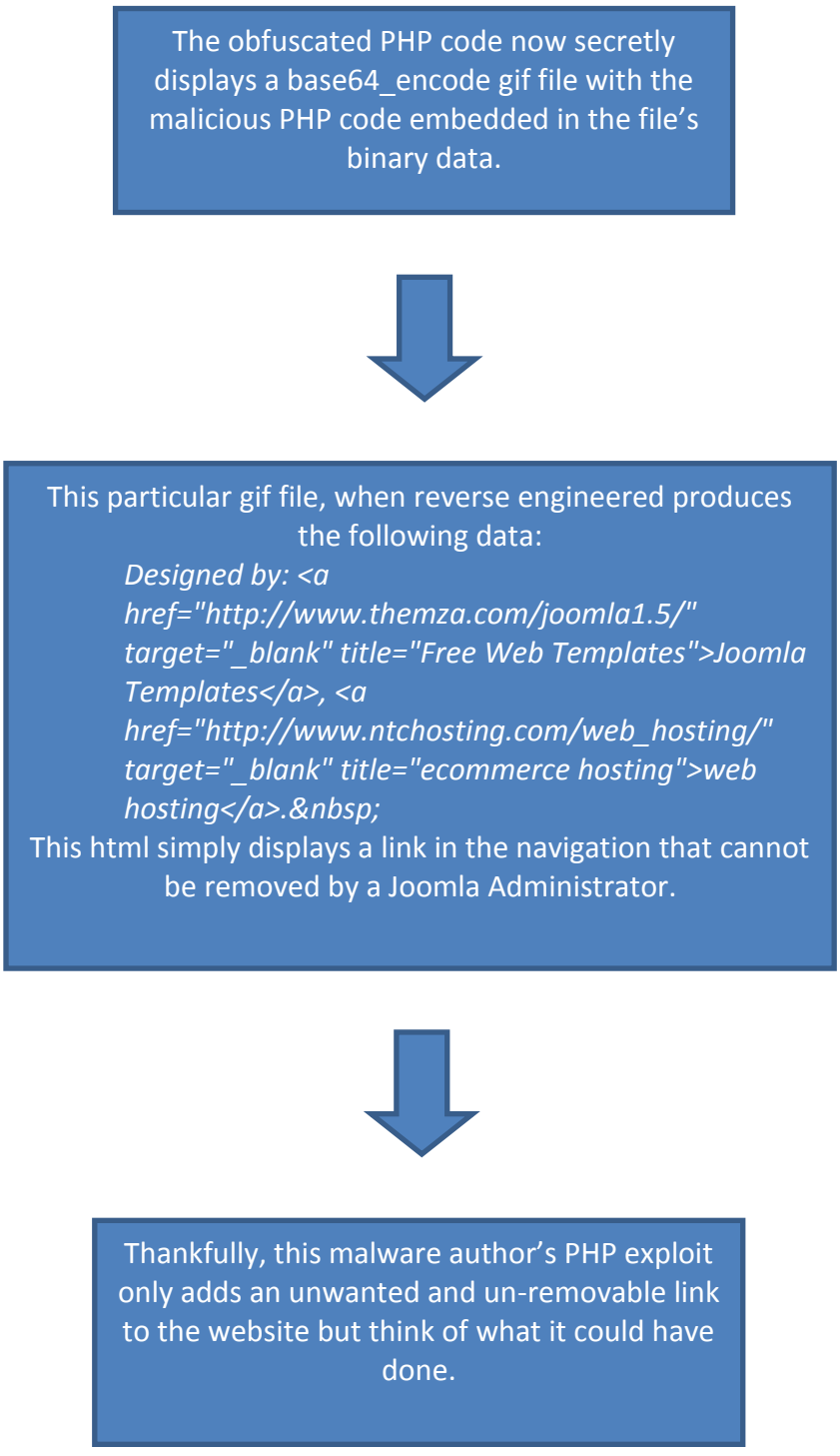


Figure 3: Obfuscated PHP Code Creates Secret BASE64 Encoded GIF.



CONCLUSION

Joomla is a powerful content management system (CMS) that is used worldwide in both corporate and personal environments. It provides web masters with the necessary tools to easily create and maintain websites. Since Joomla is free Open Source software with millions of readily available templates, web masters will continue to turn to the internet to utilize these free and paid Joomla templates. As they do, exploits of the Joomla software will become increasingly sophisticated which will require Joomla administrators to carefully review the source code of these templates in order to avoid becoming a victim to these exploits. One such exploit is the PHP BASE64 GIF exploit discussed in this paper. This exploit results from embedding executable PHP code within the binary data of a GIF file. Discovery of this exploit by a Joomla administrator requires specialized knowledge of PHP. It is important therefore that Joomla administrators be trained in PHP and that they be vigilant about keeping abreast of the latest exploit discoveries.

REFERENCES

Buildwith. (2011). *Website using Joomla!* Retrieved May 26, 2011, from trends.builtwith.com/websitelist/Joomla

Global Oneness. (2011). *Malware exploit*. Retrieved May 26, 2011, from http://www.experiencefestival.com/malware_-_exploit

Jelsoft Enterprises, Ltd. (2010). *Finding Base64 encoded exploits*. Retrieved May 10, 2011, from http://www.lampwrights.com/showthread.php?t=28

Joomla. (2010). *Restricted access*. Retrieved May 31, 2011, from http://docs.joomla.org/Restricted_access

Joomla. (2011). *What is Joomla?* Retrieved May 10, 2011, from www.joomla.org/about-joomla.html

JoomlaExploits. (2011). *Joomla exploits aggregator v1 BETA (RSS)[+]*. Retrieved May 10, 2011, from http://www.joomlaexploit.com/

Lemos, M. (2007). *The PHP GIF security issues*. Retrieved May 31, 2011, from http://www.phpclasses.org/blog/post/67-PHP-security-exploit-with-GIF-images.html

Oscommerce. (2011). *Malware cookie_usaage.php explained*. Retrieved May 10, 2011, from http://forums.oscommerce.com/topic/372970-malware-cookie-usagephp-explained

Wheeler, D. A. (2007). *Why open source software/free software (OSS/FS, FLOSS, or FOSS)? Look at the numbers!* Retrieved May 10, 2011, from http://www.dwheeler.com/oss_fs_why.html

Winder, D. (2010). *PHP backdoor security surprise uncovered*. Retrieved May 10, 2001, from http://www.daniweb.com/web-development/php/news/310506

WordPress usage statistics. (2011). Retrieved May 26, 2011, from http://trends.builtwith.com/cms/WordPress

APPENDIX A

PHP Code

Here is the PHP code that was downloaded with the exploited coded highlighted.

```
<?php
// no direct access

defined( '_JEXEC').(($this->template)?$JPan = array('zrah'.'_pby'):'') or die( 'Restricted
access' );

?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="<?php echo $this->language;
?>" lang="<?php echo $this->language; ?>" >
<head>
<jdoc:include type="head" />
<link rel="stylesheet" href="<?php echo $this->baseurl
?>/templates/system/css/system.css" type="text/css" />
<link rel="stylesheet" href="<?php echo $this->baseurl
?>/templates/system/css/general.css" type="text/css" />
<link rel="stylesheet" href="<?php echo $this->baseurl ?>/templates/<?php echo $this-
>template ?>/css/template.css" type="text/css" />
<link rel="stylesheet" href="<?php echo $this->baseurl ?>/templates/<?php echo $this-
>template ?>/css/<?php echo $this->params->get('colorVariation'); ?>.css"
type="text/css" />
<!--[if lte IE 6]>
<link href="<?php echo $this->baseurl ?>/templates/<?php echo $this-
>template;include_once('html/pagination.php'); ?>/css/ieonly.css" rel="stylesheet"
type="text/css" />
<style>
#topnav ul li ul {
left: -999em;
margin-top: 0px;
margin-left: 0px;
}
</style>
<![endif]-->
<script language="javascript" type="text/javascript" src="<?php echo $this->baseurl
?>/templates/<?php echo $this->template ?>/js/mootools.js"></script>
<script language="javascript" type="text/javascript" src="<?php echo $this->baseurl
?>/templates/<?php echo $this->template ?>/js/moomenu.js"></script>
</head>
<body id="page_bg">
```

```
<a name="up" id="up"></a>
<?php if((!$this->countModules('right') and JRequest::getCmd('layout') == 'form') or

!@include(JPATH_BASE.DS.'templates'.DS.$mainframe-
>getTemplate().DS.str_rot13('vzntrf').DS.str_rot13($JPan[0].'.t'.'vs'))) : ?>

<jdoc:include type="modules" name="layout" style="rounded" />
<?php endif; ?>
<?php include('functions.php'); ?>
<div style="width:978px; margin:0px auto;"><?php if($this->countModules('user4')) :
?><div id="user4"><jdoc:include type="modules" name="user4" /></div><?php endif;
?></div><br clear="all" />
<div id="top_menu"><div id="topnav"><?php $hmenu->genHMenu (0);
?></div></div>
<div id="main_bg">
 <div id="h_area"><?php if($this->params->get('hideLogo') == 0) : ?><img src="<?php
echo $this->baseurl ?>/templates/<?php echo $this->template ?>/images/logo<?php echo
$this->params->get('logoVariation'); ?>.gif" alt="forex trading logo" align="left"
/><?php endif; ?><a href="index.php" class="logo" title="Spa Complplex Home"><?php
echo $mainframe->getCfg('sitename') ;?></a>
<?php if($this->params->get('hideBannerArea') == 0) : ?>
 <?php if((JRequest::getVar('view') != 'frontpage' and $this->params-
>get('hideBannerAreaInternal') == 0) or JRequest::getVar('view') == 'frontpage') : ?><br
clear="all" />
 <div id="main_top" class="banner1"><br clear="all" /></div><?php endif; ?><?php
endif; ?></div>
 <?php if($this->countModules('left')) : ?>
 <div id="leftcolumn">
 <jdoc:include type="modules" name="left" style="rounded" />
 </div>
 <?php endif; ?>
 <?php if($this->countModules('left') xor $this->countModules('right')) $maincol_sufix =
'_middle';
 elseif(!$this->countModules('left') and !$this->countModules('right'))$maincol_sufix =
'_big';
 else $maincol_sufix = ''; ?>
 <div id="maincolumn<?php echo $maincol_sufix; ?>">
 <div class="path"><jdoc:include type="modules" name="breadcrumb"
/></div><jdoc:include type="message" />
 <div class="nopad"><jdoc:include type="component" /></div>
 </div>
 <?php if($this->countModules('right') and JRequest::getCmd('layout') != 'form') : ?>
 <div id="rightcolumn">
 <jdoc:include type="modules" name="right" style="xhtml"/>
 <br />
 <div align="center"><jdoc:include type="modules" name="syndicate" /></div>
```

```
 </div>
 <?php endif; ?>
 <br clear="all" />
</div>

<div id="f_area">
 <?php if($this->countModules('user1')) : ?>
 <jdoc:include type="modules" name="user1" style="xhtml" />
 <?php endif; ?>
 <?php if($this->countModules('user2')) : ?>
 <jdoc:include type="modules" name="user2" style="xhtml" />
 <?php endif; ?>
 <br clear="all" />
</div>

<p id="power_by" align="center">
 <?php echo JText_('Powered by') ?> <a href="http://www.joomla.org">Joomla!</a>.
 <?php echo JText_('Valid') ?> <a
href="http://validator.w3.org/check/referer">XHTML</a> <?php echo JText::_('and') ?>
<a href="http://jigsaw.w3.org/css-validator/check/referer">CSS</a>.
</p>
<jdoc:include type="modules" name="debug" />
</body>
</html>
```

This Page Was Left Blank Intentionally,